



# Sistema de Navegación para Robots Sociales Basado en Señales

Ana del Valle Corrales Paredes

Director:  
Dr. Miguel Ángel Salichs

Escuela Politécnica Superior  
de la Universidad Carlos III de Madrid





## **Sistema de Navegación para Robots Sociales Basado en Señales**







# Sistema de Navegación para Robots Sociales Basado en Señales

Ana del Valle Corrales Paredes

Director:  
Dr. Miguel Ángel Salichs

Tesis Doctoral

2012

Escuela Politécnica Superior  
de la Universidad Carlos III de Madrid

Universidad Carlos III de Madrid

Publication Data:

Ana del Valle Corrales Paredes

Sistema de Navegación para Robots Sociales Basado en Señales

Universidad Carlos III de Madrid

Copyright © 2012 Ana del Valle Corrales Paredes

*A mis padres Ana Ramona y Ramón,  
y a mis hermanas Nayi, Glennny y Alba.*



# AGRADECIMIENTOS

Una de las secciones que más emociona escribir al momento de redactar una tesis es sin duda la sección de los agradecimientos. El hecho de tener en mis manos el resumen y fruto de horas de trabajo y esfuerzo, me motiva para dar reconocimiento a aquellas personas que han aportado directa o indirectamente su grano de arena para que mi tesis doctoral sea un hecho.

Primeramente quiero agradecer a mi tutor, el Prof. Miguel Ángel Salichs por su orientación y dedicación para la realización de esta tesis, la cual ha sido el resultado de muchas conversaciones e intercambio de ideas. Muchas gracias.

También quiero agradecer al Dr. Ramón Barber, por la ayuda que me ha brindado durante la realización de esta tesis. Sus observaciones y consejos han sido muy útiles para alcanzar mi objetivo.

A mis compañeros del equipo “Maggie”, poco a poco hemos crecido y pasado a ser el “Social Robots Team”. Quiero agradecer principalmente a David, por toda su ayuda con los problemas de hardware, comunicación, software, por “resucitar” a Maggie... si pongo aquí la lista no termino, así que lo resumo en una frase: ¡Muchas gracias por haber vuelto!. También agradezco a Álvaro, por su disponibilidad cuando he tenido alguna duda de software, a Alberto por su ayuda en la adaptación de este trabajo a la nueva versión de AD. A Javi y Fernando por responder mis dudas con respecto al sistema de diálogos. También al resto de mis compañeros de equipo Arnaud, Víctor y María y a los que han pasado por aquí y a los que igualmente estoy agradecida. Ha sido un placer haber compartido con vosotros esta etapa académica.

Quiero agradecer a los profesores del Laboratorio de Sistemas Discretos Automatización e Integración (LaSDAI), de mi Alma Máter, la Universidad de Los Andes: Eladio Dapena y Rafael Rivas, por su ayuda y motivación para hacer mi doctorado. Agradezco también a FUNDACITE y a su entonces presidente, el Prof. José Aguilar, por el impulso económico inicial para comenzar con mi postgrado.

A mis compañeros de despacho, con los cuales he compartido innumerables horas de trabajo: Carla, Fer, Álvaro y, en su momento, Javi. Gracias por hacer más amenos los días de trabajo. Agradezco también a mis demás compañeros del Departamento ISA de la UC3M, que me han acompañado en mi etapa como investigadora y profesora de esta universidad.

A nivel personal quiero darle las gracias a mi familia, que aunque estemos en diferentes continentes, siempre han estado apoyándome y dándome motivos para seguir adelante. Dedico este logro a mis padres: Ana Ramona y Ramón, a mis hermanas: Nayi, Gleny y Alba y a mis queridos sobrinos: Eduardo, Luis, Luisany, Lupita, Ángel y Ariani. Ha sido duro estar lejos y haberme perdido algunos momentos, pero ha

tenido su recompensa.

No soy del tipo de personas que tiene muchos amigos, pero los que tengo son personas muy valiosas. Gracias especiales a Alejandra, Bea, Carla, Gustavo, Laura, Mariemma, Maritza, Nestor, Osmay, Ricardo y Teresa, por sus palabras de apoyo y aliento durante el desarrollo de esta tesis, por los buenos momentos que hemos compartido y sobre todo gracias por estar pendientes de nuestra amistad. Sé que se alegran por este logro.

Y finalmente quiero agradecer a David, por todo su apoyo, por leer de arriba a abajo esta tesis y por sus valiosas recomendaciones. Gracias por el cariño y compañía, y sobre todo por esa energía positiva que tiene cada día y hace que vea las cosas desde un punto de vista distinto. Gracias también a la familia García Godoy por ser mi familia en este lado del Atlántico.

Muchas gracias a todos.

*Ana Corrales P.*

# RESUMEN

*“Si no puedes explicarlo con sencillez,  
no lo has entendido bien.”  
(Albert Einstein)*

La autonomía de un robot es uno de los principales objetivos que los investigadores en robótica se han trazado en las últimas décadas. Lograr esto abarca un abanico muy amplio de sub-objetivos, dependiendo de la funcionalidad final del robot y las potenciales tareas que podría realizar. Así como en los humanos nos ha sido vital para nuestra supervivencia movernos por el entorno y llegar a un lugar o destino específico, para un robot móvil la navegación es una tarea primordial.

Cuando nos movemos por entornos desconocidos necesitamos saber nuestra ubicación en el contexto espacial (edificio, calle, barrio, ciudad...), la ubicación del lugar destino (o al menos la dirección en que se encuentra) y cómo ir a ese lugar. Al no tener conocimientos previos, la señalización en el entorno se convierte en un elemento primordial para alcanzar el objetivo. La señalización es el conjunto de elementos colocados en el entorno que, mediante indicaciones gráficas, auditivas o táctiles, guían y ayudan a las personas a navegar en un área o edificio. La señalización sirve de soporte para la orientación en un espacio a un lugar determinado y facilita la accesibilidad a los servicios requeridos dando seguridad en los desplazamientos y las acciones.

A menudo las personas necesitamos llegar a un sitio determinado sin tener conocimiento previo del entorno en que nos movemos, bien sea porque es la primera vez que estamos en ese lugar o porque no disponemos de un mapa que nos ayude a orientarnos. Así como los humanos debemos enfrentarnos a este tipo de situaciones, un robot autónomo que deba moverse por el entorno también debe ser capaz de resolver este problema de forma natural, usando los recursos que encuentre en el entorno para llegar satisfactoriamente a la meta.

En esta tesis se presenta un sistema de señalización con el principal objetivo de orientar e informar al robot para que éste pueda cumplir su meta de navegación. Este sistema comprende desde el diseño físico de la señal y el tipo de señales que se utilizarán hasta el algoritmo de navegación que especifica el proceso de búsqueda de las señales y su interpretación, para que el robot ejecute las acciones necesarias para alcanzar el destino asignado. También se propone un sistema de escritura de las señales del entorno, de modo que el robot pueda interactuar con las personas para adquirir la información de las señales que se quieren escribir y así automatizar el proceso de señalización.

Para la implementación del sistema de señalización se ha escogido utilizar los sistemas de Identificación por Radio Frecuencia (RFID), ya que son un buen recurso para enriquecer la información que un robot puede obtener del entorno, acelerando considerablemente el proceso de lectura de las señales sin depender de factores ambientales como la luz, además que son de bajo coste, no perjudican la transitabilidad de las personas ni la de los robots y no ejercen impacto en el entorno.

Los sistemas descritos se han integrado e implementado con éxito en el robot social Maggie del Robotics Lab de la Universidad Carlos III de Madrid.



# ABSTRACT

*“If you can’t explain it simply,  
you don’t understand it well enough.”  
(Albert Einstein)*

The autonomy of a robot is one of the main objectives that researchers in robotics have been set in the last decades. Achieving this goal includes a very wide range of subobjectives, depending on the final functionality and the potential tasks that a robot can do. Just as the movement through the environment and getting to a place or a specific destination has been vital for human survival, for a mobile robot navigation is a basic task.

When we move through unfamiliar environments we need to know our location in the spatial context (building, street, neighbourhood, city...), the location of the destination (or at least the direction in which it is) and how to go there. With no prior knowledge, signalling in the environment becomes a key element to achieve our objective. Signalling is the set of items placed in the environment that, by means of graphical displays, auditory or tactile guides help people navigate in an area or building. Signalling acts as a support for guidance in space to a given location and facilitates the accessibility of required services providing security in movements and actions.

Often, people need to go to a particular site without having prior knowledge of the environment in which we move, whether it is the first time we are in that place or if we do not have a map. Just as humans have to deal with these types of situations, an autonomous robot that moves through the environment must also be capable of solving this problem in a natural way, using the resources that it finds in the environment to satisfactorily reach the goal.

This thesis presents a signage system with the main purpose of guiding and informing the robot so that it can fulfill its navigational goal. This system begins from the physical design of the signal and the type of signals to be used to the navigation algorithm that specifies the search process signals and their interpretation, so that the robot can execute the necessary actions to reach the assigned destination. This work also proposes a system of writing the environmental signals, so that the robot can interact with people to acquire the information of the signals that must be written and thus automatize the process of signalling.

For the implementation of the signalling system we have chosen the Radio Frequency Identification systems (RFID), because these are good resources to enrich the information that a robot can get from the environment, considerably accelerating the

process of reading the signals without depending on environmental factors such as light. In addition, the tags used are low cost, do not impair the trafficability of people or robots and have no impact on the environment.

The described systems have been integrated and implemented successfully in the social robot Maggie, the Robotics Lab robot at the Carlos III University of Madrid.

# Índice general

<b>Agradecimientos</b> . . . . .	IX
<b>Resumen</b> . . . . .	XI
<b>Abstract</b> . . . . .	XIII
<b>1. Introducción</b> . . . . .	1
1.1. Motivación . . . . .	1
1.2. Definición del problema . . . . .	1
1.3. Objetivos de la tesis . . . . .	2
1.4. Estructura y organización de la tesis . . . . .	3
<b>2. El problema de la navegación</b> . . . . .	5
2.1. Introducción . . . . .	5
2.2. Navegación . . . . .	5
2.2.1. Modelos y estrategias del proceso de Wayfinding . . . . .	6
2.2.2. Sistemas de soporte de navegación para las personas . . . . .	7
2.3. Navegación y Robótica . . . . .	11
2.3.1. Tipos de Navegación . . . . .	12
2.3.2. Navegación por marcas ( <i>Landmarks</i> ) . . . . .	13
2.4. Navegación y los Sistemas RFID (Radio Frequency IDentification) . .	17
2.5. Navegación e interacción humano-robot . . . . .	23
2.6. Discusión . . . . .	28
<b>3. Señalización del entorno</b> . . . . .	29
3.1. Introducción . . . . .	29
3.2. Señalización del entorno . . . . .	30
3.2.1. El signo . . . . .	30
3.2.2. La señal . . . . .	31
Clasificación de las señales . . . . .	32
3.2.3. Señalización en Edificios . . . . .	35
3.3. Diseño del sistema de señalización para navegación con robots móviles	36
3.3.1. Diseño de las señales . . . . .	36
¿Qué tipo de información es necesaria? . . . . .	37
3.3.2. Estructura de la información . . . . .	44
3.3.3. Diseño físico de las señales. Almacenamiento de la información	45

3.4. Discusión . . . . .	47
<b>4. Sistema de navegación mediante señales . . . . .</b>	<b>49</b>
4.1. Introducción . . . . .	49
4.2. Definiciones . . . . .	50
4.3. Algoritmo de Navegación . . . . .	51
4.4. Discusión . . . . .	57
<b>5. Señalización basada en la Interacción Humano-Robot . . . . .</b>	<b>59</b>
5.1. Introducción . . . . .	59
5.2. Adquisición de la información . . . . .	60
5.2.1. Adquisición de la información mediante interacción Humano-Robot . . . . .	60
5.2.2. Adquisición de la información por deducción a través de señales previas . . . . .	64
5.3. Diseño del sistema de diálogo para la adquisición y escritura de de datos de una señal . . . . .	65
5.3.1. Definición de las gramáticas para la adquisición de información en la señalización . . . . .	66
Frases posibles . . . . .	66
Definición del lenguaje . . . . .	67
Definición Semántica . . . . .	67
5.4. Discusión . . . . .	69
<b>6. Sistema Experimental . . . . .</b>	<b>71</b>
6.1. Introducción . . . . .	71
6.2. El Robot Maggie . . . . .	71
6.2.1. Características del robot . . . . .	71
6.2.2. Arquitectura Software del Robot . . . . .	72
Definición de Habilidad . . . . .	73
6.3. Sistema RFID implementado en el robot . . . . .	74
6.3.1. Hardware RFID en el Robot Maggie . . . . .	74
Primera selección . . . . .	75
¿Y si el robot requiere escribir etiquetas RFID para almacenar información? . . . . .	77
6.3.2. Habilidades RFID . . . . .	78
Casos de uso . . . . .	79
6.3.3. Habilidad de Lectura de Datos CRFID_ReadSkill . . . . .	80
6.3.4. Habilidad de Escritura de Datos CRFID_WriteSkill . . . . .	82
6.4. Información en las etiquetas RFID . . . . .	84
6.5. Estructura Software de las Señales . . . . .	88
6.6. Habilidad de Navegación Mediante Señales . . . . .	90
6.6.1. Lectura de las señales . . . . .	90
6.6.2. Procesamiento de la información . . . . .	90
6.7. Diálogos para la adquisición de información en la señalización . . . . .	95

6.7.1.	Parte 1: <code>dialogPlaceWaymarking</code> . . . . .	96
6.7.2.	Parte 2: <code>dialogOptionWaymarking</code> . . . . .	97
6.7.3.	Parte 3: <code>dialogConnectionWaymarking</code> . . . . .	98
6.8.	Habilidad de Señalización . . . . .	100
6.8.1.	Implementación de la Habilidad de Señalización . . . . .	102
6.9.	Discusión . . . . .	108
<b>7.</b>	<b>Resultados Experimentales</b> . . . . .	<b>109</b>
7.1.	Introducción . . . . .	109
7.2.	Habilidad de Navegación Mediante Señales . . . . .	110
7.2.1.	Experimento 1 . . . . .	110
	Objetivo . . . . .	110
	Descripción . . . . .	110
	Datos de entrada: . . . . .	111
	Resultados . . . . .	113
7.2.2.	Experimento 2 . . . . .	115
	Objetivo . . . . .	115
	Descripción . . . . .	115
	Datos de entrada: . . . . .	115
	Resultados . . . . .	119
7.2.3.	Experimento 3 . . . . .	119
	Objetivo . . . . .	119
	Descripción . . . . .	119
	Datos de entrada: . . . . .	119
	Resultados . . . . .	124
7.2.4.	Experimento 4 . . . . .	124
	Objetivo . . . . .	124
	Descripción . . . . .	124
	Datos de entrada: . . . . .	124
	Resultados . . . . .	127
7.2.5.	Experimento 5 . . . . .	127
	Objetivo . . . . .	127
	Descripción . . . . .	127
	Datos de entrada: . . . . .	128
	Resultados . . . . .	135
7.3.	Señalización del Entorno . . . . .	136
7.3.1.	Definición de la Gramática Semántica . . . . .	136
7.3.2.	El robot toma la iniciativa . . . . .	137
	¿Dónde estoy? [ <code>dialogPlaceWaymarking</code> ] . . . . .	138
	Formularios utilizados – Diálogo en <code>voiceXML</code> . . . . .	139
	Solicitud de información adicional [ <code>dialogOptionWaymarking</code> ] . . . . .	140
	¿Hacia donde puedo ir? [ <code>dialogConnectionWaymarking</code> ] . . . . .	141
	Formularios utilizados – Diálogo en <code>voiceXML</code> . . . . .	144
7.3.3.	El usuario toma la iniciativa . . . . .	146
	El usuario indica al robot el lugar actual . . . . .	146

El usuario indica al robot acciones de movimiento . . . . .	147
Gramática y formularios utilizados – Diálogos en <i>voiceXML</i> . .	148
7.3.4. Escritura de la información . . . . .	151
7.4. Discusión . . . . .	153
<b>8. Conclusiones y trabajos futuros . . . . .</b>	<b>155</b>
8.1. Principales Aportaciones . . . . .	156
8.2. Trabajos Futuros . . . . .	157
8.3. Comentarios Finales . . . . .	158
 <b>Apéndice . . . . .</b>	 <b>159</b>
<b>A. Herramienta para la construcción de mapas geométricos . . . . .</b>	<b>161</b>
A.1. SLAM: Simultaneous Localization and Mapping . . . . .	161
A.2. PMAP Particles Mapping . . . . .	162
<b>B. Sistemas de Identificación por Radio Frecuencia . . . . .</b>	<b>165</b>
B.1. Ventajas y Limitaciones de los sistemas RFID . . . . .	168
<b>C. Habilidades Básicas para Navegación . . . . .</b>	<b>169</b>
C.1. Habilidad de Localización Geométrica . . . . .	169
C.1.1. Cómo Localizar el Robot Geométricamente en el Robot Maggie	170
C.2. Habilidad “ <i>Ir a un punto</i> ” . . . . .	173
C.2.1. Control Cinemático de la Posición . . . . .	173
C.2.2. Comportamiento de la Habilidad “ <i>Ir a un punto</i> ” . . . . .	177
C.3. Habilidad Seguir Contorno . . . . .	181
C.3.1. Control Proporcional para la habilidad <i>Seguir contorno</i> . . . .	181
C.4. Habilidad Evitar Obstáculos . . . . .	185
C.4.1. Habilidad de Detección de Obstáculos . . . . .	185
C.4.2. CSkill_SafeGoToPoint . . . . .	186
Casos posibles . . . . .	186

# ÍNDICE DE TABLAS

6.1. Rango de Frecuencias RFID. . . . .	75
6.2. Etiquetas RFID según su alimentación. . . . .	75
6.3. Etiquetas según el tipo de memoria. . . . .	75
C.1. Posibles valores que indican la zona o zonas donde se han detectado obstáculos. . . . .	186





# Índice de figuras

1.1. Señalización que encontramos en el Aeropuerto de Barajas - Madrid.	2
2.1. Taxonomía del proceso de Wayfinding [Wiener et al., 2009]. . . . .	8
2.2. Sistema de Navegación implementado en [Liu et al., 2009], funcionando sobre un móvil Nokia N95. . . . .	9
2.3. Elementos del sistema <i>Tactile Wayfinder</i> . . . . .	9
2.4. Comportamiento del sistema <i>Tactile Wayfinder</i> . El cono indica la zona de estimulación táctil en el cinturón. . . . .	10
2.5. Talking Points 3: mecanismo de selección de las notificaciones al usuario basado en las distancias de ubicación de los <i>PdIs</i> . . . . .	11
2.6. Ejemplo de navegación local: cuando la meta es visible a nuestro sistema de percepción (Fotografía de Google Maps - ©2012 Google). . . .	12
2.7. Ejemplo de navegación global: cuando la meta no es visible a nuestro sistema de percepción (Fotografía de Google Maps - ©2012 Google). . . .	12
2.8. Ejemplo de una Marca Visual: Edificio Metrópolis de Madrid – España.	14
2.9. Ejemplo de una Marca Estructural. . . . .	14
2.10. Ejemplo de una Marca Semántica: Frutería en un barrio de Ciudadela – España. . . . .	14
2.11. Marcas en el entorno con contenidos de texto. . . . .	15
2.12. Segmentos de líneas reconocidos como marcas del entorno. . . . .	16
2.13. Reconocimiento de luces fluorescentes como marcas. . . . .	16
2.14. Prototipo del código MR. . . . .	16
2.15. 3 IRIDs y un robot móvil [Lee and Song, 2007]. . . . .	17
2.16. Códigos de barras invisibles usados para navegación. . . . .	17
2.17. Robot de Tmsuk y etiquetas RFID usadas. . . . .	18
2.18. Robot y elementos de hardware usados para la orientación con RFID.	19
2.19. Localización de la etiqueta RFID. . . . .	20
2.20. Robot Guía RG-I y etiquetas pasivas usadas. . . . .	21
2.21. Grafo de conectividad usado en el RG-I. . . . .	22
2.22. iWalker. . . . .	23
2.23. Ejemplo de ruta: las líneas azules representan las etiquetas RFID con sus identificadores. . . . .	24
2.24. Robot SIGA, desarrollado por YDreams. . . . .	24
2.25. Robot ACE (Autonomous City Explorer) interactuando con los peatones.	25
2.26. Sistema Navspace. Las indicaciones las proporciona el usuario a través del teclado. . . . .	26

2.27. A la izquierda la maqueta de ciudad realizada para que navegue un robot humanoide. A la derecha el mapa topológico generado usando <i>FRI</i> s. . . . .	27
3.1. Elementos que intervienen en un proceso semiótico. . . . .	31
3.2. Mapa de la Rosaleda Cecilio Rodriguez, Parque de El Retiro - Madrid, con la señalización <i>Usted está aquí</i> . . . . .	32
3.3. Ejemplo de señal informativa: Horario de trenes en la estación de Atocha (Madrid). . . . .	33
3.4. Ejemplo de señales identificativas: Hotel Santander y el Teatro Reina Victoria - Madrid. . . . .	33
3.5. Ejemplo de señales direccionales: Señal ubicada cerca de la Universidad Carlos III. . . . .	34
3.6. Ejemplo de señal de advertencia: Atención, suelo mojado. . . . .	34
3.7. Mapas de las zonas geográficas para definir el lugar Barrio de Chamberí. . . . .	38
3.8. Mapa del Aeropuerto Internacional de Barajas en Madrid. . . . .	39
3.9. Representación del <i>lugar</i> Edificio Betancourt de la UC3M y el conjunto de zonas anidadas al que pertenece. . . . .	40
3.10. Plano de la Escuela Politécnica Superior. Campus de Leganés UC3M. . . . .	40
3.11. Definición del <i>lugar</i> : Mapa de la tercera planta del Edificio Betancourt de la UC3M. . . . .	41
3.12. Código alfanumérico para la identificación de despachos y salas en la UC3M. . . . .	41
3.13. Señalización en el Aeropuerto Internacional de Barajas en Madrid. . . . .	42
3.14. Edificio Betancourt de la UC3M, y algunas conexiones. . . . .	43
3.15. Factores que intervienen en el diseño de un sistema de señalización. . . . .	45
3.16. Etiquetas RFID pasivas. (Fotografía de <a href="http://www.rfideas4u.com/">http://www.rfideas4u.com/</a> ) . . . . .	46
4.1. Proceso de orientación y búsqueda de la meta en entornos desconocidos. . . . .	49
4.2. Representación de un lugar $l_0$ y una conexión $c_i$ . . . . .	51
4.3. Representación de una señal. . . . .	52
4.4. Caso 1: La meta coincide con el lugar que señala la etiqueta. . . . .	52
4.5. Caso 2: La meta coincide con una conexión. . . . .	53
4.6. Señalización hacia una zona genérica . . . . .	54
4.7. Caso 3: La meta coincide con una zona de una conexión . . . . .	54
4.8. Señal genérica indicando la salida . . . . .	55
4.9. Diagrama de actividades del algortimo de navegación . . . . .	56
5.1. Ejemplo del proceso de adquisición de información mediante deducción a través de una señal previa: Robot navegando por un pasillo a través de dos señales. . . . .	65
6.1. El Robot Social Maggie. . . . .	72
6.2. Arquitectura Automático-Deliberativa AD. . . . .	73
6.3. Esquema de comunicación entre habilidades en la Arquitectura AD. . . . .	74
6.4. Lector RFID de la empresa FEIG Electronic y etiquetas pasivas. . . . .	76

6.5. Etiquetas HF en el entorno. . . . .	77
6.6. Lector UHF RFID de la empresa FEIG Electronic. . . . .	78
6.7. El robot Maggie con las antenas UHF. . . . .	78
6.8. Diagrama de flujo de datos de la habilidad de lectura RFID. . . . .	80
6.9. Diagrama de flujo de datos de la habilidad de escritura de etiquetas RFID. . . . .	81
6.10. Arquitectura software básica del sistema RFID. . . . .	81
6.11. Diagrama de comportamiento de la habilidad de escritura de datos. .	83
6.12. Diagrama de Clases de la Habilidad de Lectura de Etiquetas RFID. .	84
6.13. Diagrama de Clases de la Habilidad de Escritura de Etiquetas RFID. .	85
6.14. Representación de un <i>lugar</i> : <i>Laboratorio de Maggie</i> . . . . .	89
6.15. Representación de una <i>conexión</i> . . . . .	89
6.16. Diagrama de flujo de la Habilidad de Navegación Mediante Señales. .	91
6.17. Diagrama de clases de la Habilidad de Navegación Mediante Señales. .	93
6.18. Mapa geométrico del laboratorio 1.3.C.12. . . . .	95
6.19. Diagrama de actividades: <code>dialogPlaceWaymarking</code> . . . . .	96
6.20. Diagrama de actividades: <code>dialogOptionWaymarking</code> . . . . .	98
6.21. Diagrama de actividades: <code>dialogConnectionWaymarking</code> . . . . .	99
6.22. Esquema del sistema de señalización y sus diferentes elementos. . . .	100
6.23. Diagrama de actividades de la Habilidad de Señalización <code>CSkill_Waymarking</code> . .	103
6.24. Grafo de dependencias de la Habilidad de Señalización <code>CSkill_Waymarking</code> . . . . .	104
7.1. Robot Maggie, donde se ha integrado e implementado el sistema de navegación . . . . .	109
7.2. Plano de parte de la Tercera Planta del Edificio Betancourt de la Universidad Carlos III de Madrid . . . . .	110
7.3. Objetivo 1: El robot parte desde el punto indicado por la flecha y deberá llegar hasta la meta asignada, indicada con una estrella. . . .	110
7.4. Posición inicial del robot y posibles acciones que indica la señal al robot. .	111
7.5. Lugar que indica la señal inicial. . . . .	111
7.6. Lugares hacia donde puede ir el robot y las acciones que debe realizar para ese fin. . . . .	112
7.7. Experimento 1: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot . . . . .	113
7.8. Experimento 1: Información que lee el robot de las señales en la tarea de navegación. En verde se indica el lugar a donde debe ir el robot y en naranja la acción que ejecuta. . . . .	114
7.9. Objetivo 2: El robot parte desde el punto indicado por la flecha y deberá llegar hasta la meta asignada, indicada con una estrella. . . .	115
7.10. Lugar que indica la señal inicial. . . . .	116
7.11. Lugares hacia donde puede ir el robot y las acciones que debe realizar para ese fin. . . . .	116
7.12. Experimento 2: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot . . . . .	117

7.13. Experimento 2: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot . . . . .	117
7.14. Experimento 2: Información que lee el robot de las señales en la tarea de navegación. En verde se indica el lugar a donde debe ir el robot y en naranja la acción que ejecuta. . . . .	118
7.15. Objetivo 3: El robot parte desde el punto indicado por la flecha y deberá llegar hasta la meta asignada, indicada con una estrella. . . .	119
7.16. Posición inicial del robot y posibles movimientos que puede realizar para buscar señales en el entorno. . . . .	120
7.17. El robot explora el entorno hasta encontrar una señal. . . . .	120
7.18. Información que lee el robot al encontrar la primera señal. . . . .	121
7.19. Experimento 3: Información que lee el robot de las primeras señales que encuentra en la tarea de navegación. . . . .	122
7.20. Experimento 3: Resumen de la información de las señales 1.3.C03 a la 1.3.C.11 con las acciones que realiza el robot. . . . .	123
7.21. Experimento 3: Recorrido realizado por el robot. . . . .	124
7.22. Experimento 4: Información de las señales que encuentra el robot. . .	126
7.23. Experimento 4: Recorrido realizado por el robot y su comportamiento al encontrar un obstáculo que impide la lectura de una señal. . . . .	127
7.24. Lugares hacia donde puede ir el robot y las acciones que debe realizar para ese fin. . . . .	128
7.25. Identificador del punto destino y de la zona genérica que toma en cuenta el robot. . . . .	129
7.26. Experimento 5: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot hasta llegar al Laboratorio 1.3.C.12 . .	129
7.27. Experimento 5: Información que lee el robot de las señales en la tarea de navegación. En verde se indica el lugar a donde debe ir el robot y en naranja la acción que ejecuta. . . . .	130
7.28. Experimento 5: Ubicación de las señales en el Laboratorio 1.3.C.12 . .	131
7.29. Experimento 5: Información de la señal situada en el Laboratorio 1.3.C.12	131
7.30. Mapa geométrico del Laboratorio 1.3.C.12. El robot no está localizado geométricamente. . . . .	132
7.31. Robot localizado geométricamente en la entrada del Laboratorio 1.3.C.12	133
7.32. Robot localizado geométricamente frente al Televisor del Laboratorio 1.3.C.12 . . . . .	133
7.33. Experimento 5: Plano del entorno con las señales (etiquetas RFID) y el recorrido realizado por el robot. . . . .	135
7.34. Primer caso de señalización: El robot toma la iniciativa en la interacción.	137
7.35. Ejemplo de señalización del entorno. El usuario da órdenes de movimiento al robot. . . . .	151
A.1. Mapa realizado con pmap en la USC . . . . .	163
A.2. Gráfico del error absoluto: odometría vs pmap . . . . .	163
A.3. Laboratorio 1.3C14 y pasillo 3ra planta Edif. Betancourt UC3M usando pmap . . . . .	163

B.1. Elementos de un sistema RFID . . . . .	165
B.2. Principio Maestro-Eslavo entre la aplicación software, lector y etiquetas. . . . .	166
B.3. Tipos de Polarización de las Antenas RFID . . . . .	167
B.4. Etiquetas RFID . . . . .	167
C.1. Diagrama de clases de la habilidad de Localización Geométrica. . . .	171
C.2. Imagen del mapa del Laboratorio de Robótica con el robot no localizado.	172
C.3. Imagen del mapa del Laboratorio de Robótica con el robot localizado.	173
C.4. Modelo cinemático del robot para la habilidad Ir a un Punto . . . . .	174
C.5. Controlador Proporcional-Derivativo para la habilidad “Ir a un punto”	175
C.6. Cambio de rango del ángulo de $-\pi$ a $\pi$ . . . . .	176
C.7. Diagrama de actividades de la habilidad <i>Ir a un Punto</i> . . . . .	178
C.8. Grafo de dependencias de la habilidad <i>Ir a un Punto</i> . . . . .	179
C.9. Diagrama de herencia de la habilidad <i>Ir a un Punto</i> . . . . .	180
C.10. Controlador Proporcional para la habilidad “Seguir Contorno” . . . .	181
C.11. Modelo para la habilidad “Seguir Contorno” . . . . .	182
C.12. División por zonas del rango de visión del láser. . . . .	185
C.13. División de las zonas donde pueden ser detectados los obstáculos cercanos al robot. . . . .	187
C.14. Casos en que se activará la habilidad “Seguir contorno” por la derecha del robot. . . . .	188
C.15. Casos en que se activará la habilidad “Seguir contorno” por la izquierda del robot. . . . .	189
C.16. Casos en que se activará la habilidad “Ir a un punto”. . . . .	189



# 1. INTRODUCCIÓN

*“Life is one big road with lots of signs.”  
(Bob Marley)*

## 1.1. Motivación

Una de las tareas mas comunes a la que nos enfrentamos los humanos consiste en movernos por el entorno y llegar a un lugar o destino específico, aunque no lo hayamos visitado anteriormente, no tengamos conocimientos previos del entorno o no contemos con un mapa. Este proceso que incluye la orientación, búsqueda y seguimiento de un camino hacia una meta determinada, es un proceso cognitivo que ha sido ampliamente estudiado en campos como la psicología [Tolman, 1948, Golledge, 1999], antropología [Istomin and Dwyer, 2009] y geografía [Lynch, 1960] y es conocido con el término de “*Wayfinding*”, que podríamos traducir como “búsqueda del camino” o “autoguiado”.

Existen situaciones cotidianas que nos motivan a que realicemos este proceso, por ejemplo, si estamos en un aeropuerto y tenemos que llegar a la puerta de embarque, buscaremos en nuestro billete el número de la puerta destino, e inmediatamente buscaremos en la señalización existente el número que corresponda a la puerta de embarque como las que vemos en la figura 1.1. Cuando vamos a visitar a un amigo al hospital y buscamos su habitación, inicialmente preguntamos al personal de recepción por el destino y le pediremos que nos indique cómo llegar para posteriormente ir guiándonos por los paneles informativos que encontremos en la puerta de las habitaciones. Otro caso distinto es cuando visitamos un edificio público por primera vez, generalmente nuestra primera acción es buscar un mapa global y localizar el sitio al que queremos ir.

Como podemos ver, si no tenemos conocimiento previo del camino a seguir, la información existente en el entorno es la que nos ayuda a llegar al destino, la información espacial puede ser la señalización del entorno o en su defecto la información verbal que nos puedan proporcionar personas que encontremos a nuestro alrededor. Así como los humanos debemos enfrentarnos a este tipo de situaciones, un robot autónomo que deba moverse por el entorno también debe ser capaz de resolver este problema de forma natural y así llegar satisfactoriamente a la meta asignada.

## 1.2. Definición del problema

La autonomía de un robot es uno de los principales objetivos que los investigadores en robótica se han trazado en las últimas décadas. Lograr ésto abarca un abanico muy



*Fig. 1.1: Señalización que encontramos en el Aeropuerto de Barajas - Madrid.*

amplio de sub-objetivos, dependiendo de la funcionalidad del robot y las potenciales tareas que podría realizar. Así como en los humanos el proceso de *Wayfinding* ha sido vital para su supervivencia en el entorno, para un robot móvil la navegación es una tarea primordial. Numerosas investigaciones están orientadas al estudio y diseño de diferentes técnicas y métodos para lograr la navegación autónoma en un robot móvil, sin embargo muchas de estas técnicas no emulan el procesamiento cognitivo de los humanos, es por esto que se presenta el reto de que un robot móvil pueda emular el comportamiento de los humanos al momento de diseñar estrategias para la navegación.

La mayoría de los sistemas de navegación implementados en robots móviles, requieren de información previa (como mapas) para que el robot pueda guiarse y moverse por el entorno para llegar a la meta; en caso contrario, el robot debe moverse por todo el entorno y realizar un mapa moviéndose y localizándose simultáneamente. En la actualidad existen muy pocos trabajos que enfoquen la navegación en robótica de manera más natural y que use la información espacial que hay en el entorno tal y como lo hacemos los humanos.

Así como los humanos debemos enfrentarnos a este tipo de situaciones, un robot que deba moverse por entornos estructurados también debe ser capaz de resolver este problema usando los recursos que encuentre en el entorno para llegar satisfactoriamente a la meta. Este problema abarca aspectos fundamentales como la adquisición de la información (sensores), modelado de la información, interpretación, procesamiento y finalmente la ejecución de acciones para alcanzar el punto destino.

### 1.3. Objetivos de la tesis

El objetivo principal de esta tesis es el diseño y desarrollo de un **sistema de navegación basado en señales**, para un robot móvil que se mueve en un entorno desconocido.



Para este fin, es necesario alcanzar los siguientes objetivos parciales:

- Diseñar un sistema de señalización con el principal objetivo de orientar e informar al robot para que éste pueda cumplir su meta de navegación. Este diseño comprende desde el diseño físico de la señal y el tipo de señales que se utilizarán hasta el contenido informativo de éstas.
- En cuanto a la información que las señales contendrán, es necesario modelar los elementos del entorno que el robot tomará en cuenta para la tarea de la navegación.
- Diseñar un algoritmo de navegación que especifique el proceso de búsqueda de las señales y su interpretación, para que el robot ejecute las acciones necesarias para alcanzar el destino asignado.
- Implementar el algoritmo en un robot social real, esto implica diseñar y poner en marcha el sistema sensorial que adquirirá la información de las señales. Es por esto que también se requiere realizar un estudio de la tecnología existente para incorporar al robot el o los sensores necesarios para la detección de las señales.
- Realizar un sistema de escritura de señales del entorno. El robot al encontrar una señal, podrá agregar y/o modificar la información contenida en la señal, con la finalidad de automatizar el proceso de señalización o de mejorar la información para uso propio en el futuro o para otros robots que naveguen en el entorno.
- Para realizar el proceso de señalización, el robot debe adquirir la información que escribirá en la señal. Con este fin se debe agregar al sistema la interacción humano-robot, específicamente la comunicación verbal, ya que con este tipo de comunicación, es posible intercambiar información más específica y enriquecedora a la hora de escribir y/o complementar la información de las señales de navegación en el entorno.

## 1.4. Estructura y organización de la tesis

El contenido de esta tesis está conformado por ocho capítulos. Inicialmente se presenta un breve estudio del problema de la navegación, partiendo en principio de los sistemas de soporte para la navegación en las personas para seguidamente estudiar el problema de la navegación en la robótica, citando algunos trabajos relacionados con esta tesis. El capítulo 3 muestra el diseño de la señalización del entorno para posteriormente en el capítulo 4 centrarnos en el sistema de la navegación basada en señales. En el capítulo 5 se explica el sistema de escritura y modificación de las señales conjuntamente con la integración de diálogos para la interacción humano-robot. Seguidamente en el capítulo 6, se presenta el sistema experimental, donde explicaremos como se ha implementado el algoritmo de navegación, el robot utilizado y el sistema sensorial integrado para llevar a cabo este trabajo. Los resultados experimentales del

sistema de navegación basado en señales son presentados en el capítulo 7 así como el sistema de escritura de las señales por parte del robot. Finalmente, en el capítulo 8 concluimos con los aportes principales de esta tesis, algunos comentarios finales sobre el trabajo realizado y los trabajos futuros. Adicionalmente se han anexado un conjunto de apéndices, que ayudarán al lector a complementar su visión de esta tesis, donde, entre otras cosas, se ha incluido una breve explicación de los elementos de software que se han desarrollado en este trabajo para el correcto funcionamiento del sistema en el robot.

## 2. EL PROBLEMA DE LA NAVEGACIÓN

### 2.1. Introducción

La autonomía de un robot es uno de los principales objetivos que los investigadores en robótica se han trazado en los últimas décadas. George Bekey [Bekey, 2005], define robots autónomos como:

*Máquinas inteligentes capaces de realizar, por sí mismos, tareas en el entorno, sin control humano explícito.*

Las tareas autónomas que debe realizar un robot, varían dependiendo de su funcionalidad, bien sea si es un robot móvil, humanoide, aéreo o submarino (entre otros). Una de las tareas básicas para la autonomía, consiste en moverse en entornos desconocidos y llegar a un destino determinado, tal y como lo hacemos los humanos.

En los humanos el proceso de orientación, búsqueda y seguimiento de un camino o ruta hacia un destino a una meta determinada es denominado “**Wayfinding**”, que podríamos traducir como “búsqueda del camino” o “autoguiado”. En robótica, el término utilizado es “**Navegación**”.

En robótica móvil diferentes técnicas y métodos se han y se están desarrollando para lograr la navegación autónoma, el reto de los investigadores en robótica consiste en emular el comportamiento de los humanos al momento de diseñar estrategias para navegación. En este capítulo definiremos los conceptos mas importantes en el área de navegación en robótica móvil, así como un resumen general del estado del arte de las diferentes estrategias de navegación relacionadas con este trabajo. En la sección 2.4, nos centramos en los trabajos de navegación usando la tecnología RFID, ya que esta tecnología es una parte importante en la implementación de esta tesis. Posteriormente en la sección 2.5 se comentarán diferentes trabajos donde la interacción humano-robot toma importancia como soporte en los sistemas de navegación.

### 2.2. Navegación

En la bibliografía existe una variedad de definiciones para el concepto *Navegación*, todos ellos se resumen en la siguiente definición:

Conjunto de acciones que se realizan para llegar a una meta asignada o destino desde un punto de origen determinado.

Según [Montemello, 2005], la navegación está compuesta por dos procesos: *wayfinding* (autoguiado o búsqueda del camino) y *locomoción*. *Wayfinding* resuelve los problemas de orientación, toma de decisiones, y selección de rutas mientras que la *locomoción* se refiere a la coordinación de los movimientos (usando el sistema sensorial y motor), con el fin de avanzar hacia la meta de manera segura.

El término *wayfinding* fue originalmente introducido en 1960 por el urbanista Kevin Lynch [Lynch, 1960], al referirse al conjunto de elementos que usamos las personas cuando nos sentimos desorientados y buscamos un camino para llegar a un destino determinado, por ejemplo: mapas, señales, carteles, nombres de calles, números de portales, etc. Lynch definió el término **Wayfinding** como el :

*Uso y organización de los elementos del entorno que nos permiten navegar con éxito a través de espacios complejos.*

Es un concepto generalmente usado en los campos de psicología y geografía, para definir el proceso en el que se utilizan recursos del espacio para nuestra orientación, búsqueda y seguimiento de un camino o ruta desde un origen o lugar de partida a una meta o destino determinado.

Numerosas teorías han tratado de explicar el modo en que las personas se ubican, localizan y llegan a un destino. Según [Golledge, 1999], este proceso esta compuesto de un conjunto de procesos cognitivos de adquisición, codificación, almacenamiento, decodificación y uso de la información.

### 2.2.1. Modelos y estrategias del proceso de Wayfinding

Existen diferentes propuestas o modelos de representación del proceso de Wayfinding.

Allen [Allen, 1999] distingue 3 categorías, clasificadas dependiendo del nivel de familiaridad o conocimiento del entorno:

- La meta corresponde a un destino conocido, como ir de casa al trabajo.
- Exploración. No hay una meta concreta, simplemente consiste en explorar el entorno y volver al punto de origen, por ejemplo, si nos mudamos de casa y comenzamos a explorar los alrededores.
- La meta corresponde a un destino desconocido, una de las tareas más comunes a la que nos enfrentamos cuando visitamos nuevos sitios.

Así mismo, el autor expone los siguientes métodos para aplicar en las diferentes categorías de Wayfinding mencionadas:

- Orientado a la búsqueda. Es útil cuando se está realizando la exploración del entorno, el individuo se orienta en el camino de ida usando la información que percibe y se reorienta al volver al punto de origen conocido. Sin embargo, también puede ser usado para ir a destinos conocidos o desconocidos. La principal desventaja es que sólo es eficiente cuando la distancia a recorrer es corta, el entorno es estático y se emplea un método de búsqueda sistemático de la meta.

- Siguiendo un camino previamente marcado. Este método puede ser aplicado para los 3 tipos de Wayfinding, la desventaja es el coste de señalizar todo el camino.
- Navegar a través de puntos de referencia (o *landmarks*). El individuo se basa en una secuencia de puntos de referencia que le proporcionan la información del entorno.
- Locomoción habitual. Sucede cuando se ha automatizado el proceso de ir de un punto a otro, debido a la repetición de la tarea. Funciona en entornos y destinos conocidos. Por ejemplo, el camino de casa al trabajo.
- Usando un mapa cognitivo. Las personas forman una representación interna del entorno, comprendiéndolo de forma coherente y predecible a través del uso de mapas mentales formados por los siguientes elementos: caminos, arcos, distritos, nodos y marcas o puntos de referencia.

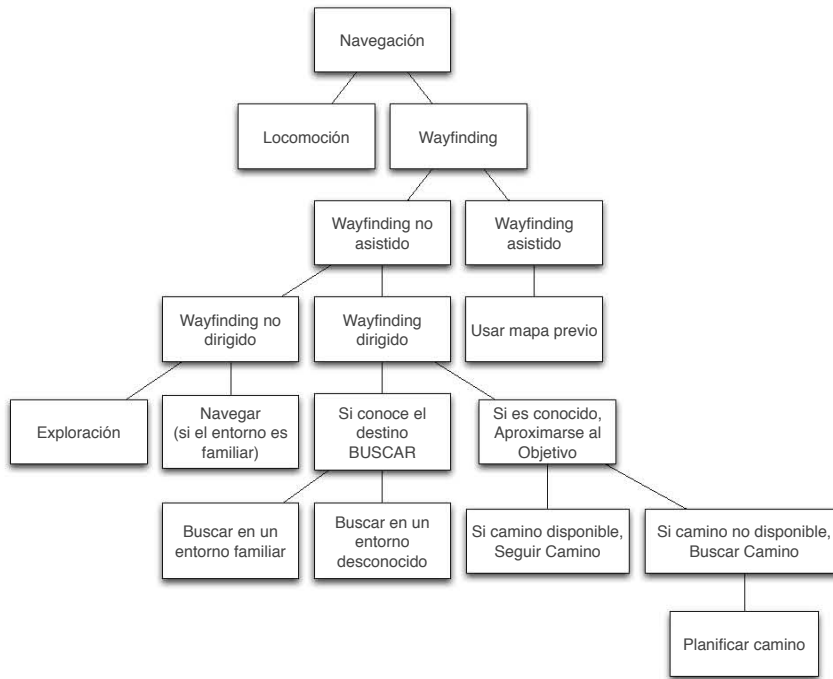
En [Xia et al., 2008], los autores presentan un estudio de los tipos de *wayfinding* basado en el comportamiento de los turistas al visitar un parque. La finalidad de este estudio es facilitar al turista los recursos necesarios para que puedan visitar el parque sin problema alguno. Con esta finalidad propone 4 modelos de representación del proceso de Wayfinding basados en los expuestos por Allen.

- Wayfinding cuando se está familiarizado con el entorno.
- Wayfinding cuando se está familiarizado parcialmente con el entorno.
- Wayfinding cuando se desconoce el camino que se debe tomar.
- Wayfinding cuando se desconoce el camino que se debe tomar pero existen puntos de referencia en el entorno.

En [Wiener et al., 2009] los autores presentan una taxonomía de un sistema de navegación para distinguir los diferentes tareas en el proceso de *wayfinding* en los humanos. En este artículo se argumenta que el conocimiento sobre la ubicación de una meta específica, el conocimiento de una ruta específica hacia esa meta, y el conocimiento del entorno, son aspectos fundamentales para determinar la estrategia que se puede aplicar para resolver una tarea de navegación. El esquema que presenta el autor, está basado en las diferencias existentes entre wayfinding asistido y wayfinding sin ayudas externas del entorno, también especifica los diferentes niveles de conocimiento del entorno para cada caso (Fig. 2.1).

### 2.2.2. Sistemas de soporte de navegación para las personas

En la bibliografía podemos encontrar diferentes estudios sobre sistemas de navegación para las personas, la mayoría de éstos están orientados a personas de la tercera edad, personas con discapacidad cognitiva y personas visualmente incapacitadas.



**Fig. 2.1:** Taxonomía del proceso de Wayfinding [Wiener et al., 2009].

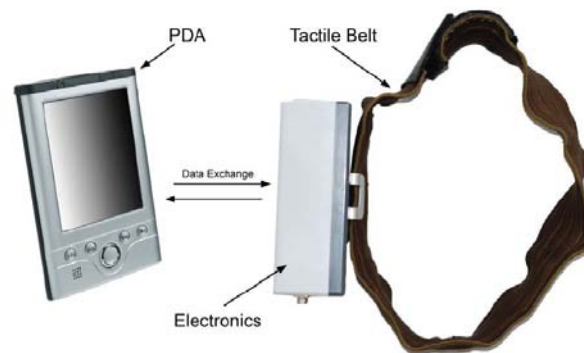
En [Liu et al., 2009], los autores han diseñado y desarrollado un sistema automatizado para generar en tiempo real instrucciones de navegación en entornos exteriores, a través de un dispositivo móvil. El sistema genera indicaciones de dos tipos: indicaciones basadas en marcas naturales del entorno e indicaciones de movimiento (como “continúa hacia adelante” o “gira a la derecha”). Además posee un proceso de selección de marcas naturales por lo que dispone de una base de datos con fotografías geo-etiquetadas (con la información de la marca y sus coordenadas GPS). Las indicaciones relacionadas con las marcas son generadas por un sistema de decisión usando un modelo de Markov (Markov Decision Process), éste está basado en el usado para planificación de caminos con incertidumbre implementado en el campo de la robótica [Thrun et al., 2005]. En la figura 2.2, se muestra el sistema de navegación implementado en un teléfono móvil, la interfaz gráfica muestra una imagen del entorno próximo con la indicación de movimiento que debe realizar el usuario. Además muestra en texto la indicación al igual que por audio. El móvil debe tener activo el GPS para el correcto funcionamiento del sistema, por lo que sólo funciona en entornos exteriores.

Otras investigaciones han orientado la interacción de los sistemas de navegación usando el sentido del tacto.

En [Heuten et al., 2008], han desarrollado un sistema para dar soporte al usuario en la navegación usando interacción táctil. El sistema denominado *Tactile Wayfinder*, está compuesto por un cinturón con vibradores, que indica al usuario las direcciones y las desviaciones o giros en la trayectoria y una PDA como interfaz para asignar el punto de inicio y fin de la misión de navegación y como conexión entre el sistema planificador y el cinturón (Ver Fig. 2.3).



**Fig. 2.2:** Sistema de Navegación implementado en [Liu et al., 2009], funcionando sobre un móvil Nokia N95.



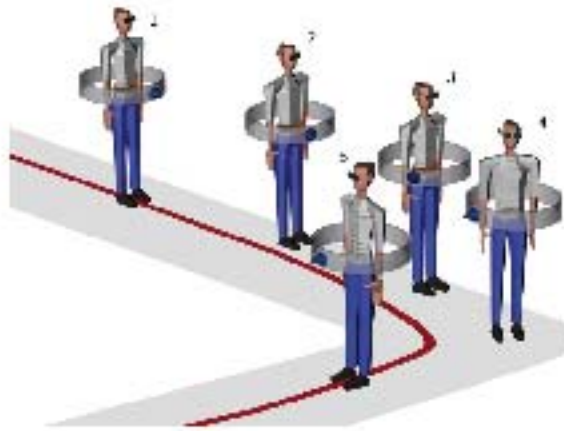
**Fig. 2.3:** Elementos del sistema Tactile Wayfinder.

Los autores dividen el proceso de Wayfinding o Navegación en 3 pasos:

1. El primer paso corresponde a que el usuario se localice en el entorno, para ésto usan el sistema de localización por GPS, lo que limita el sistema a su uso exclusivo en entornos exteriores.
2. El segundo paso corresponde a la selección de la ruta, ésto lo realiza con un algoritmo de planificación cuya entrada son los puntos de origen y destino y un mapa del entorno.
3. La tercera parte, en la que está enfocado el trabajo, se refiere a mantener al usuario sobre su ruta y evitar que se pierda. Para este fin, el sistema orienta al usuario a través de un cinturón con vibración en la piel guiándole hacia la dirección donde debe ir.

En la figura 2.4, se muestra un ejemplo de funcionamiento del sistema. En el primer paso, el viajero está en el camino correcto, así que el cinturón vibra en la parte delantera. En el paso 3 el usuario se desvía de la ruta, entonces la estimulación táctil del cinturón se sitúa en la parte derecha, indicando al usuario que esa es la dirección correcta.

Otros sistemas han sido diseñados para el soporte a la navegación para personas con discapacidad visual parcial, dándoles información a través de audio.



**Fig. 2.4:** Comportamiento del sistema *Tactile Wayfinder*. El cono indica la zona de estimulación táctil en el cinturón.

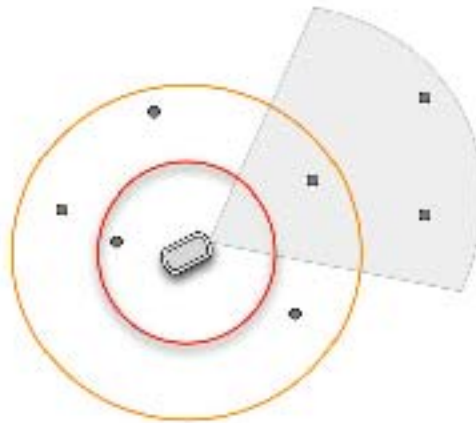
En [Yang et al., 2011], se describe un sistema de soporte a la navegación denominado *Talking Points 3 (TP3)* implementado en un smartphone Nexus One. El objetivo de este trabajo consiste en aumentar la legibilidad del entorno para facilitar la tarea de navegación de las personas en un entorno desconocido. Este sistema provee información sobre los Puntos de Interés cercanos que el usuario encuentra en el entorno a medida que va navegando. Los Puntos de Interés (o *PdI*), especifican información física del entorno (como obstáculos y marcas representativas) y además su ubicación geométrica. Estos puntos se encuentran almacenados previamente en una base de datos, que proporciona la información al sistema a medida que el usuario va navegando. El sistema TP3 proporciona al usuario 3 tipos de notificaciones dependiendo de la zona donde detecte el *PdI*:

1. Notificación automática. Notifica a los usuarios la existencia de *PdIs* en una distancia de 3 metros desde su posición actual. La notificación consiste en un sonido de alerta y, usando *text-to-speech*, la información del nombre del *PdI* y la distancia a éste.
2. Lugares cercanos. El usuario puede visualizar en el móvil una lista de *PdIs* ubicados en una distancia de 9 metros respecto a su posición actual.
3. Buscador direccional. El usuario puede apuntar a una dirección en la pantalla del teléfono y el sistema listará los *PdIs* ubicados en ángulo de 45 grados hasta una distancia de 30 metros.

La figura 2.5, muestra un esquema de funcionamiento y posición de los *PdIs*.

El principal inconveniente en la implementación del TP3, es el error en el cálculo de la distancia desde el usuario (portador del smartphone) a los puntos de interés, ya que en entornos interiores no es posible usar el GPS, para eso usa un sistema de triangulación por WiFi, que, según sus experimentos, muestra un error de  $\pm 5$  metros.





**Fig. 2.5:** *Talking Points 3: mecanismo de selección de las notificaciones al usuario basado en las distancias de ubicación de los PdIs.*

## 2.3. Navegación y Robótica

El problema de la navegación en robótica ha sido ampliamente estudiado. Todo robot que deba moverse de un sitio a otro requiere de un sistema de navegación que cumpla con la tarea de ir eficientemente desde un punto inicial a un destino final.

En [Salichs and Moreno, 2000] los autores plantean que un sistema de navegación debería cumplir con el siguiente objetivo:

*Se coloca un robot en un entorno desconocido, amplio, complejo y dinámico. Una vez que el robot ha explorado el entorno, éste debe ser capaz de ir a cualquier lugar del entorno, tratando de minimizar una función de coste (por ejemplo: tiempo, energía, etc.).*

Para cumplir este objetivo es necesario que el robot pueda realizar diferentes acciones para responder a las preguntas que plantea el problema de la navegación [Bekey, 2005]:

- *¿Cómo llego al destino?*, esto implica diferentes tareas: avanzar sin chocar con obstáculos del entorno, buscar y/o usar información existente, buscar el mejor camino, minimizar costos.
- *¿Donde estoy?*, esto se refiere al problema de localización. Saber donde se encuentra respecto a un punto de referencia conocido o a un sistema de coordenadas.
- *¿Donde he estado?*, esto se refiere al problema de localización y construcción de mapas de manera simultanea, por ejemplo el SLAM (*Simultaneous Localization and Mapping*).

### 2.3.1. Tipos de Navegación

En [Salichs, 2001], el autor clasifica la navegación dependiendo de la distancia en la que se encuentre la meta y si está dentro del rango de visión del robot.

**Navegación Local** Es la navegación que se realiza en el área de alcance de medición del sistema de percepción (p.ej.: Láser, sensores sonar, visión, etc.). Los objetos y la meta final se encuentran dentro del rango de percepción del robot (Fig. 2.6).



**Fig. 2.6:** Ejemplo de navegación local: cuando la meta es visible a nuestro sistema de percepción (Fotografía de Google Maps - ©2012 Google).

**Navegación Global** Es la navegación que se realiza fuera del área de alcance de medición del sistema de percepción. La meta final no se encuentra dentro del rango de percepción del robot (Fig. 2.7).



**Fig. 2.7:** Ejemplo de navegación global: cuando la meta no es visible a nuestro sistema de percepción (Fotografía de Google Maps - ©2012 Google).

Otros autores [Murphy, 2000], clasifican la navegación dependiendo del tipo de información que use el robot para moverse por el entorno:

**Navegación Topológica** También llamada *Navegación Cualitativa*, es la aproximación más natural para modelar un sistema de navegación en un robot. Toma como referencia características importantes del entorno o puntos de referencia (*landmarks*). Por ejemplo: *Sigue el pasillo y entra por la primera puerta a la derecha*. Generalmente, la representación del conocimiento del entorno es modelado mediante grafos de conectividad.

**Navegación Geométrica** También denominada *Navegación Cuantitativa*. Este modelo de navegación está basado en medidas geométricas. La posición del robot, los elementos del entorno y la meta se definen por sus coordenadas. El modelo del entorno es representado mediante mapas geométricos y/o mapas de rejillas.

**Navegación Topo-Geométrica** O *Navegación Híbrida*, combina información geométrica simple e información simbólica. El modelo del entorno representa la conectividad entre elementos geométricos y simbólicos [Dapena, 2002].

El desarrollo de esta tesis está enfocado en el uso de la navegación topológica, es por eso que en la siguiente sección se ampliará su estudio, más específicamente la navegación por marcas, y se mostrarán algunas investigaciones relacionadas.

### 2.3.2. Navegación por marcas (*Landmarks*)

Las marcas son rasgos característicos del entorno o puntos de referencia, que un robot puede reconocer a través de sus sensores. Estos pueden ser simples formas geométricas (p. ej. círculos, rectángulos), características del entorno (p.ej, esquinas, ventanas, puertas), y pueden incluir información adicional (p.ej: códigos de barras, etiquetas de Radio Frecuencia, etc.) [Borenstein et al., 1994].

En modelos topológicos y relacionales, la representación del entorno generalmente es modelada mediante grafos de conectividad denominados *Mapas Topológicos*. Los nodos son interpretados como lugares, regiones, posiciones o marcas del entorno. Los arcos representan la conexión entre los nodos, que bien pueden representar el camino transitable u órdenes de control de movimiento.

Sorrows y Hirtle [Sorrows and Hirtle, 1999], clasifican las marcas del entorno en tres categorías:

1. **Marcas Visuales:** Son referencias visuales características del entorno. Pueden distinguirse fácilmente aún si se es o no familiar al entorno (Fig. 2.8 ).
2. **Marcas Estructurales:** Son aquellas marcas en las que se busca especificar claramente la localización en el entorno (Fig. 2.9).
3. **Marcas Semánticas:** Son las marcas en las que, más que por su forma, son relevantes por su significado o por su rol en el entorno. Por ejemplo, la puerta de la oficina de inscripciones del campus, la frutería del barrio, etc. (Fig. 2.10).



*Fig. 2.8: Ejemplo de una Marca Visual: Edificio Metrópolis de Madrid – España.*



*Fig. 2.9: Ejemplo de una Marca Estructural.*



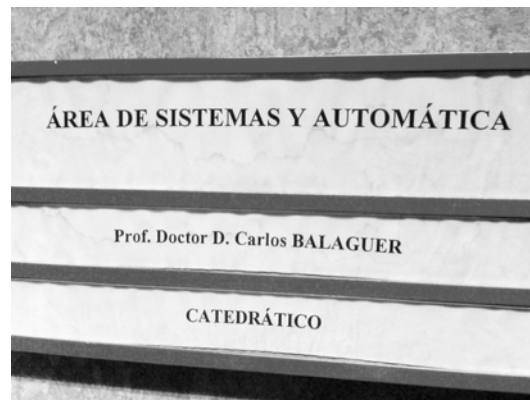
*Fig. 2.10: Ejemplo de una Marca Semántica: Frutería en un barrio de Ciudadela – España.*

Según la naturaleza de las marcas en el entorno, éstas se pueden clasificar en Marcas Naturales y Marcas Artificiales:

1. **Marcas Naturales:** son los objetos o características del entorno que ya están en el medio ambiente y tienen una función distinta a la de navegación del robot. El principal problema de la navegación con este tipo de marcas es la detección e identificación de sus características usando los sensores del robot.

Diferentes proyectos han desarrollado sistemas de navegación con marcas naturales, estos se basan en el procesado de la información de sensores como cámaras (visión) y sensores láser.

En [Mata et al., 2001] presentan un sistema de percepción visual y reconocimiento de marcas naturales y artificiales como herramienta para la localización. Básicamente reconocen marcas con contenidos de texto o iconos en lugares diseñados para uso humano en interiores y exteriores (p.ej: placas de identificación en la entrada de despachos) (ver Fig. 2.11).



**Fig. 2.11:** Marcas en el entorno con contenidos de texto.

Otros autores usan cámaras estéreo para el reconocimiento de características del entorno. En [Moon et al., 1999] proponen un método de selección de marcas naturales constituidos por segmentos de líneas verticales sobre una superficie plana (ver Fig.2.12). Otro ejemplo lo podemos ver en [Wang and Ishimatsu, 2005], aquí los autores presentan un sistema de navegación autónoma para una silla de ruedas, en este trabajo las luces fluorescentes de techo son elegidas como marcas con la ventaja de que no requieren una instalación adicional (ver Fig. 2.13).

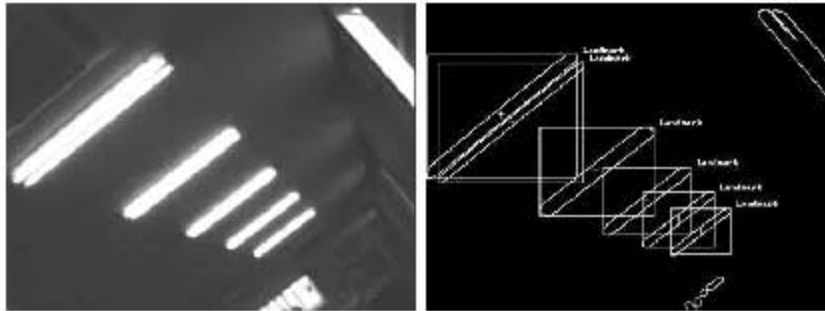
2. **Marcas Artificiales:** son marcas u objetos especialmente diseñados que deben ser colocadas en el medio ambiente con el único fin de permitir la navegación del robot.

En la literatura podemos encontrar diversos sistemas de detección de marcas de navegación artificiales mediante visión. Por ejemplo, en [Wen et al., 2009], presentan un sistema de navegación de robots móviles en interiores, haciendo uso de una marca artificial denominada *código MR (Mobile Robot)* (Fig. 2.14).

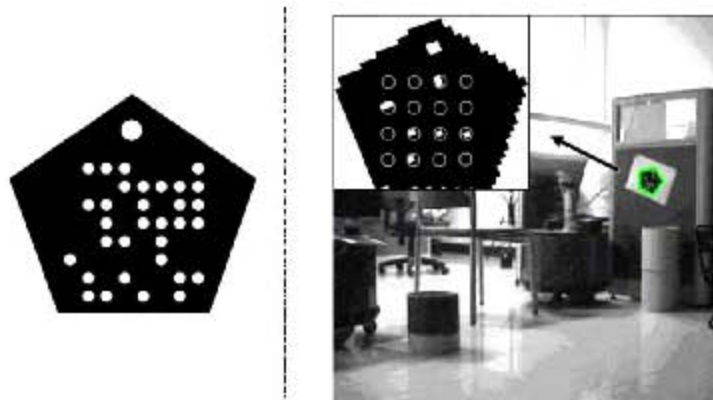




**Fig. 2.12:** Segmentos de líneas reconocidos como marcas del entorno.



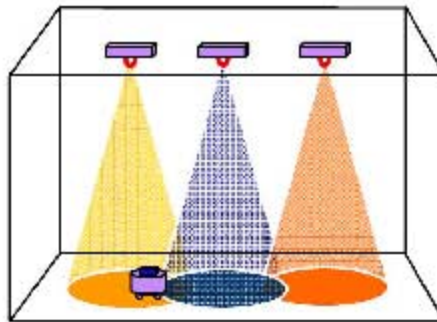
**Fig. 2.13:** Reconocimiento de luces fluorescentes como marcas.



**Fig. 2.14:** Prototipo del código MR.

Además de marcas visuales, otros trabajos de investigación han usado otro tipo de marcas. En [Lee and Song, 2007], usan luz infrarroja como marcas para localizar el robot. El sistema denominado *IRID* (*InfraRed Identification*), es implementado con LEDs IR y fototransistores. Colocan en el techo varios LED infrarrojos, el piso es dividido en sectores y cada sector tiene un único identificador. La luz infrarroja codificada indica el sector donde se encuentra el robot (cada LED IR puede emitir un único ID), el principal inconveniente en este

sistema es que si el tamaño del sector es grande, crece la incertidumbre (Fig. 2.15).



**Fig. 2.15:** 3 IRIDs y un robot móvil [Lee and Song, 2007].

En [Huh et al., 2006] usan códigos de barras 2D invisibles colocados en el suelo. La localización se realiza usando la información almacenada en los códigos una vez que ya se haya realizado el mapa del entorno (Fig. 2.16).



**Fig. 2.16:** Códigos de barras invisibles usados para navegación.

## 2.4. Navegación y los Sistemas RFID (Radio Frequency IDentification)

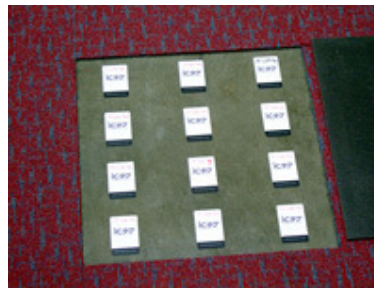
Los sistemas RFID en robótica son una herramienta innovadora para enriquecer la información que un robot puede obtener del entorno, acelerando considerablemente el proceso de identificación de objetos o personas. Además, es capaz de servir de gran soporte a otros sistemas sensoriales de un robot, por ejemplo la identificación mediante visión.

Dada la capacidad de los sistemas RFID de escribir y almacenar información en las etiquetas, un robot puede estar equipado con un lector y una antena para guardar datos que se consideren útiles en la tarea que realice, además de poder consultarla y reescribirla cuantas veces sea necesario. La información que puede obtener el robot

dependerá de lo que se desee guardar en la etiqueta, desde datos del estado físico de un objeto cualquiera, hasta saber su localización exacta.

La última tendencia de la tecnología RFID en robótica móvil se orienta a la navegación. La idea es que el robot pueda leer información de las etiquetas para saber donde se encuentra (localización), o con un conjunto de etiquetas marcar el camino que debe seguir.

- En Japón, NTT Communications<sup>1</sup> and Tmsuk<sup>2</sup> experimentaron con un robot móvil que usa RFID para navegación en un centro comercial (Figura 2.17). NTT Communications está desarrollando una tecnología básica (*m2m-x*), que enlaza redes IPv6<sup>3</sup> con tecnología RFID, con el objetivo de expandir su rango de aplicación. Para el experimento se colocaron alrededor de 5500 etiquetas RFID debajo de la alfombra, cada etiqueta posee un único código llamado EPC (Electronic Product Code). La información de la posición actual del robot y la información de la tienda que visita se encuentra en un servidor externo, el robot obtiene estos datos a través de la red IPv6. El servidor analiza la información y da instrucciones al robot para moverse a su destino final [Sho, 2006, IPv6style, 2006].



**Fig. 2.17:** Robot de Tmsuk y etiquetas RFID usadas.

- La compañía alemana Vorwerk Teppichwerke<sup>4</sup>, ha desarrollado un nuevo producto denominado *smart-floor system*, una especie de alfombra que posee etiquetas RFID pasivas embebidas, para frecuencias de 13.56 MHz. Hay cuatro etiquetas por metro cuadrado, las cuales pueden ser programadas para la navegación de sistemas automáticos de transporte, tales como robots móviles que cumplan tareas de limpieza.

El robot puede enlazar las etiquetas leídas en un *mapa virtual* del entorno y puede moverse a través de su *red de rutas* y navegar. Además guarda en memoria las etiquetas que ha detectado por lo que puede analizar los sitios por donde

<sup>1</sup> NTT Communications Global Website <http://www.ntt.com>

<sup>2</sup> Tmsuk Robots <http://www.tmsuk.co.jp/english/>

<sup>3</sup> IPv6. Cisco Systems [http://www.cisco.com/web/strategy/government/usfed\\_ipv6.html](http://www.cisco.com/web/strategy/government/usfed_ipv6.html)

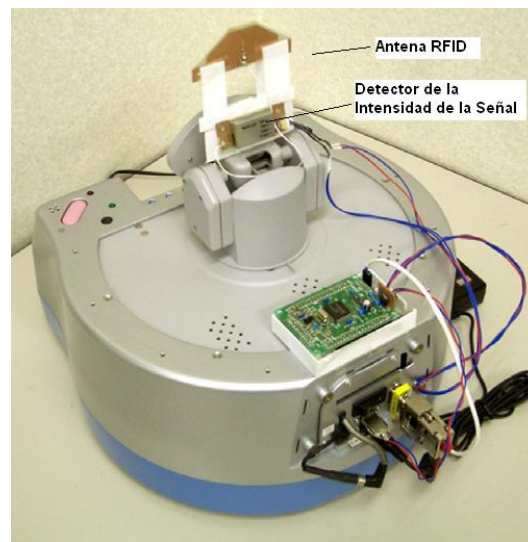
<sup>4</sup> Vorwerk presents RFID smart floor <http://www.vorwerk-teppich.de>



ha pasado o no. Si está realizando tareas de limpieza puede saber que parte de la habitación no ha limpiado o que habitaciones ya están limpias, marcando cada habitación como *limpia* o *no limpia*, teniendo así el control de la tarea que realiza.

En el caso de encontrar obstáculos en el ambiente, el robot marca en las etiquetas los obstáculos fijos y actualiza su *mapa virtual*.

- En [Kim and Chong, 2007], utilizan etiquetas RFID como elementos de orientación para el robot. Cuando una señal de RF es emitida por una etiqueta, el sistema RFID del robot recibe la información almacenada y simultáneamente envía datos sobre la dirección de la señal, esto se logra midiendo la intensidad de la señal recibida.

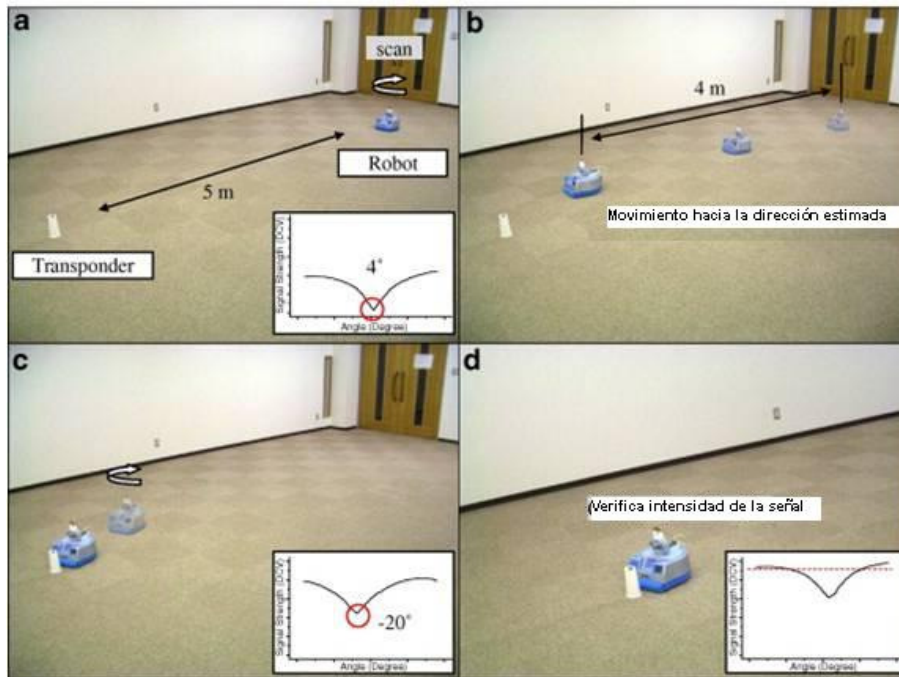


**Fig. 2.18:** Robot y elementos de hardware usados para la orientación con RFID.

Para realizar este sistema de localización, incorporaron etiquetas RFID activas de 315 Mhz, debido a que poseen mayor rango de transmisión de la señal (entre 10 y 15 metros). Además se usó una antena direccional y un detector de la intensidad de la señal, para calcular el ángulo de la señal correctamente mediante triangulación (Figura 2.18).

La localización de la etiqueta RFID es determinada mediante dos parámetros: La dirección del transponder y la distancia calculada por la intensidad de la señal. El robot necesita medir la intensidad de la señal intermitentemente mientras se mueve a una distancia estimada hasta que detecte un límite de intensidad de la señal, que indica que ha encontrado la etiqueta.

En la figura 2.19, se muestra el proceso de búsqueda y localización de la etiqueta RFID, el robot navega en dirección de la señal detectada y dependiendo de la intensidad de ésta. En la figura 2.19a, la dirección de la intensidad de la señal es  $4^\circ$  y es relativamente débil. La figura 2.19b, muestra que el robot gira  $4^\circ$  y



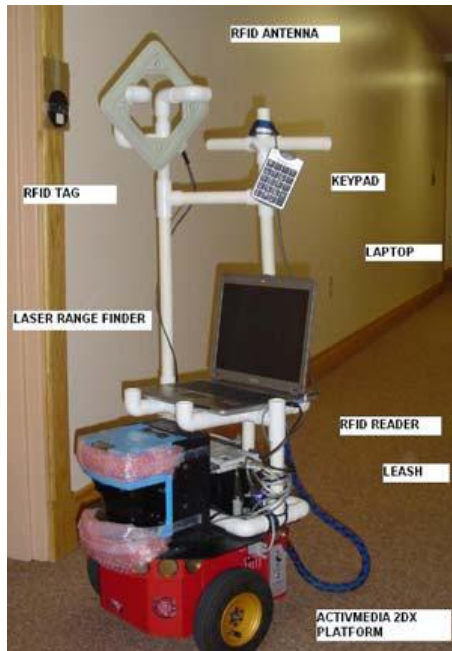
**Fig. 2.19:** Localización de la etiqueta RFID.

avanza 4 metros en dirección de la señal, dejando un margen de 1 metro. El robot posteriormente mide la señal de nuevo y calcula la distancia al transponder, el robot gira nuevamente corrigiendo errores de posición ( $20^\circ$ ) y se mueve a la nueva dirección. Finalmente en la figura 2.19d, el robot detecta el nivel umbral de intensidad de la señal, y se aproxima a la etiqueta en un rango de 50 cm.

En relación al desarrollo de sistemas de soporte para navegación en robótica usando la tecnología RFID, la bibliografía es limitada al respecto, en general la mayoría de los trabajos se refieren al uso de RFID para localización con respecto a un mapa previo. Las aplicaciones que encontramos generalmente son para soporte a personas discapacitadas o personas de la tercera edad.

- En el trabajo de Willis y Helal [Willis and Helal, 2005], han desarrollado un sistema de soporte para las personas visualmente discapacitadas. Usan una rejilla de etiquetas RFID colocadas en el suelo. Cada etiqueta es pre-grabada con la información de las coordenadas (referentes a un mapa geométrico previamente conocido) y una corta descripción del entorno. La capacidad de almacenamiento de las etiquetas RFID es limitado por lo que aplican un modelo de datos para describir los atributos del lugar. El usuario tendrá una PDA y además una pequeña antena (colocada en la planta de su zapato) para leer la información que le suministre el sistema <sup>5</sup>.
- En la Universidad de Utah [Kulyukin et al., 2006], han implementado un sistema similar. El sistema consiste de un robot móvil y etiquetas RFID pasivas

<sup>5</sup> SkyModule M1-mini HF RFID Reader Module <http://goo.gl/KSmPx>



**Fig. 2.20:** Robot Guía RG-I y etiquetas pasivas usadas.

colocadas en entornos interiores. El robot se encarga de guiar a la persona señalando en que parte del edificio se encuentran y lo lleva al destino que desee, es capaz de detectar en que habitación se encuentra y realizar planificación de caminos usando como nodos los tags RFID.

El prototipo fue evaluado en dos entornos interiores y se asume que el robot opera en entornos estructurados, teniendo en cuenta pasillos, paredes, puertas, cruces e intersecciones y obstáculos fijos. El robot guía RG-I (Figura 2.20), es un robot construido sobre una base Pioneer 2DX, posee una antena RFID TI Series 2000 de Texas Instrument, cuyo rango de lectura es 150 cm aprox., las etiquetas RFID usadas son pasivas.

El proceso de guía con RFID cumple los siguientes requisitos:

1. Cada etiqueta es programada con un único identificador.
2. Cada puerta del entorno es marcada con una etiqueta.
3. Cualquier objeto fijo en el entorno puede ser un destino final, por ejemplo, una máquina expendedora de refrescos.
4. Cada cambio en el entorno es marcado con una etiqueta (ej. una intersección).

La información y la conexión de las etiquetas es realizada *previamente y manualmente*. Representada a través de un grafo de conectividad, muestra la información del ambiente donde se moverá el robot. Los nodos representan las etiquetas RFID y los arcos el comportamiento del robot (Figura 2.21). En el grafo *f*, *u*, *l* y *r* denotan *follow-hallway* (seguir pasillo), *make-u-turn* (girar),





**Fig. 2.22:** *iWalker*.

El prototipo presentado tiene pre-codificadas las rutas por el entorno (no realiza planificación) y requiere de un mapa previo del entorno. Las rutas son definidas como una secuencia de nodos, donde cada nodo corresponde a una alfombra en el entorno. La figura 2.23 muestra un ejemplo de ruta en un entorno interior.

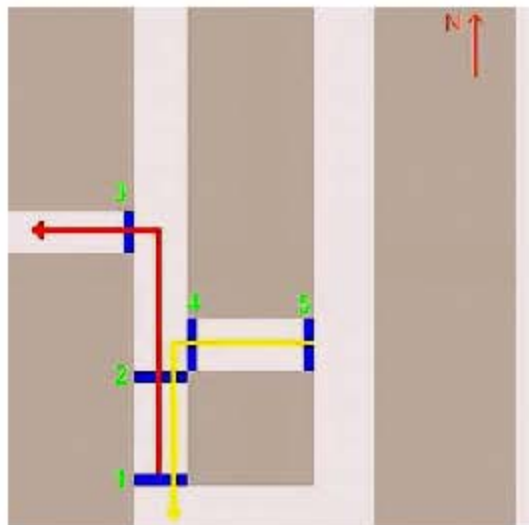
- La empresa YDreams<sup>6</sup>, ha desarrollado para el Centro de Visitantes El Faro del Banco Santander, un sistema de guía para el visitante a través de robots móviles autónomos denominados *SIGA* (*Santander Interactive Guest Assistants*) (Figura 2.24).

Cada robot utiliza dos etiquetas de radiofrecuencia (RF) que se comunican con 12 antenas RF instalados en el centro de visitantes. La información que suministra el sistema RFID en conjunto con la información de odometría y la que suministra su giróscopo, hace que los robots sitúen con exactitud su posición y orientación en el entorno. 16 sónares montados en su carcasa le permite además detectar los objetos. Un tótem externo con pantalla táctil colocado en la entrada del centro, funciona como una consola-interfaz que se comunica con los robots y los controla vía Wi-Fi. La consola permite a los visitantes seleccionar su idioma y destino, y luego ordena a uno de los robots acercarse, saludar a los visitantes y guiarlos a su destino. Una vez allí, estos robots retornan a su punto de inicio. Para este sistema es necesario un mapa previo del entorno.

## 2.5. Navegación e interacción humano-robot

En los humanos, es común pedir información a otras personas del entorno cuando queremos llegar a un sitio y no encontramos otros medios en el entorno que nos guíen

<sup>6</sup> YDreams <http://www.ydreams.com/>



**Fig. 2.23:** Ejemplo de ruta: las líneas azules representan las etiquetas *RFID* con sus identificadores.



**Fig. 2.24:** Robot *SIGA*, desarrollado por YDreams.



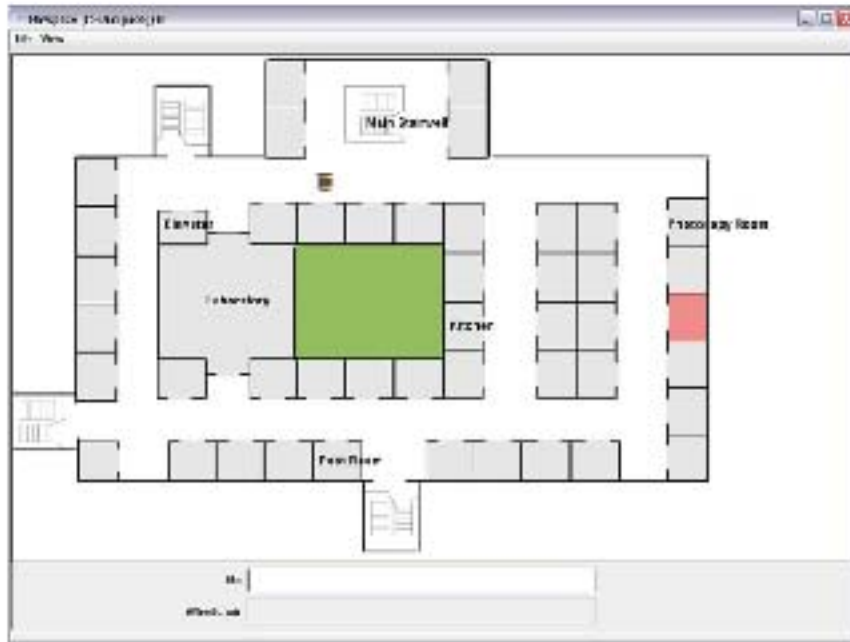
hacia el punto destino. En robótica, existen investigaciones que usan la comunicación humano-robot para que un robot pueda alcanzar satisfactoriamente la meta establecida para la navegación, tal es el caso del proyecto ACE (Autonomous City Explorer) [Weiss et al., 2010], donde un robot transita por entornos exteriores y, haciendo preguntas a los peatones, llega satisfactoriamente a su meta. Este robot usa interacción basada en voz, gestos y comunicación mediante una pantalla táctil. Para obtener información, usa expresiones del tipo : “*Hola, mi nombre es ACE, busco el camino hacia Marienplatz. Por favor, mírame a los ojos y señala la dirección que debería tomar ahora*”. En este trabajo la comunicación robot-humano es verbal, mientras que la comunicación humano-robot es gestual, por lo que la cantidad de información que el humano puede proporcionar al robot sobre el lugar a donde puede ir se limita a señalar mediante un gesto la dirección.



**Fig. 2.25:** Robot ACE (Autonomous City Explorer) interactuando con los peatones.

En [Tenbrink et al., 2010], los autores han diseñado un sistema de diálogo para dar instrucciones de ruta a un robot. Basan su estudio realizando una comparación entre la interacción humano-humano y la interacción humano-robot. En éste artículo los autores pretenden analizar el nivel y calidad de información (granularidad) que un usuario espera de una máquina para que le oriente a su destino en la tarea de navegación y la información que un usuario puede proporcionar a una máquina, en este caso un robot móvil, para que éste llegue a un punto destino. En su investigación concluyen que si bien, al dar instrucciones de orientación a un humano se trata de alcanzar un nivel de información más amplio, es decir, mayor nivel de detalle, a una máquina el usuario prefiere dar instrucciones básicas de movimientos como “gira a la derecha” o “continúa avanzando”, de modo que las instrucciones dadas a una máquina especifiquen los movimientos paso a paso durante la navegación. En la figura 2.26 se muestra la interfaz gráfica que han usado en el estudio.

En [Elmogy et al., 2008] y [Elmogy et al., 2009], los autores generan un mapa topológico a partir de un lenguaje de instrucciones al que han denominado *Formal Route Instructions* (FRIs). Cada FRI, provee instrucciones elementales en un lenguaje for-



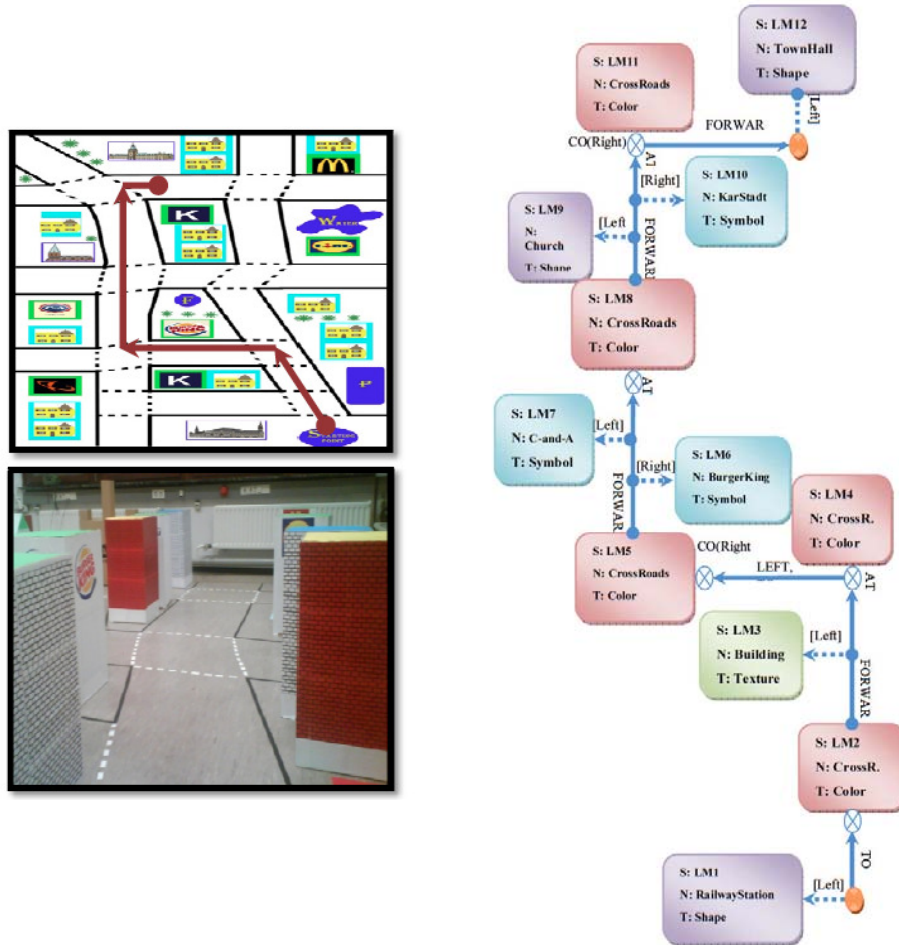
**Fig. 2.26:** Sistema Navspace. Las indicaciones las proporciona el usuario a través del teclado.

mal que internamente en el sistema del robot, son interpretadas como instrucciones de navegación. Cada instrucción FRI está relacionada con una acción, movimiento o con una marca del entorno. En las FRIs, hay tres tipos de comandos:

1. Comandos de posición, que se refiere a la posición actual del robot usando como referencia una marca del entorno. Sirven también para identificar el punto de origen y el punto destino en la tarea de navegación.
2. Comandos de locomoción. Estos comandos dan al robot órdenes de movimiento a una dirección específica o a una región del entorno, siempre tomando como referencia una o varias marcas del entorno.
3. Cambios de orientación. Estos comandos indican al robot cambios de dirección o giros específicos.

En la figura 2.5 se observa el mapa topológico generado mediante las *FRIs* y el entorno fabricado para que un robot humanoide ejecute la misión.





**Fig. 2.27:** A la izquierda la maqueta de ciudad realizada para que navegue un robot humanoide. A la derecha el mapa topológico generado usando FRIs.

## 2.6. Discusión

En este capítulo, se ha presentado un resumen de los conceptos básicos necesarios para lograr una mejor comprensión del problema de la navegación en robótica. Inicialmente hemos partido haciendo un breve estudio del problema de la navegación en las personas (Wayfinding), para posteriormente realizar una analogía entre cómo resuelven el problema de navegación las personas y cómo los investigadores en robótica están trabajando para solucionarlo en los robots. A lo largo del capítulo se han revisado algunos trabajos relacionados con la navegación en robots móviles, específicamente usando la navegación topológica que es la que más se aproxima a la que hacemos las personas. Como se ha dicho en el Capítulo 1, en esta tesis pretendemos que un robot navegue usando como soporte las señales del entorno tal y como lo hacemos los humanos al encontrarnos en un entorno desconocido y sin información previa. Para hacer un sistema que emule el comportamiento de navegación, es necesario estudiar el tipo de información que requiere el robot y el tipo de sensor que, según la tecnología actual, es acorde con el objetivo que nos hemos planteado, ya que, como sabemos, el sistema sensorial que pueda tener un robot es diferente al de los humanos. Así mismo, hemos visto que existen escasas aplicaciones que usen la interacción humano-robot para fusionar la información que un robot puede obtener de una persona en un sistema de navegación, ésto sin duda es un impulso adicional para la realización de este trabajo.

## 3. SEÑALIZACIÓN DEL ENTORNO

### 3.1. Introducción

A menudo las personas necesitamos llegar a un sitio determinado sin tener conocimiento previo del entorno en que nos movemos, bien sea porque es la primera vez que estamos en ese lugar o porque no disponemos de un mapa que nos ayude a orientarnos. Esta situación puede darse si estamos en un aeropuerto y tenemos que llegar a la puerta de embarque, cuando vamos a visitar a un amigo al hospital y buscamos su habitación o si vamos a un despacho en un edificio que desconocemos. En estos casos la información existente en el entorno es la que nos ayuda a llegar al destino, la información espacial puede ser la señalización del entorno o en su defecto la información verbal que nos pueda proporcionar personas que encontremos a nuestro alrededor.

Así como los humanos debemos enfrentarnos a este tipo de situaciones, un robot que deba moverse por entornos estructurados también debe ser capaz de resolver este problema y así llegar satisfactoriamente a la meta.

En el capítulo 2 hemos visto diversos sistemas de navegación en robótica móvil, en la mayoría de estos el robot necesita información previa para moverse por el entorno y llegar a la meta, como mapas geométricos y topológicos. En su defecto, el robot debe moverse por todo el entorno y realizar un mapa para moverse y localizarse simultáneamente. En la actualidad existen muy pocos trabajos que enfoquen la navegación en robótica de manera más natural y que use la información espacial que hay en el entorno tal y como lo hacemos los humanos.

En este capítulo presentamos un modelo de navegación en un robot móvil para la búsqueda de un destino, sin conocimiento previo del entorno. De los posibles recursos de información: señalización del entorno o información verbal, esta tesis se enfoca en el uso de la *señalización* para que el robot cumpla con el objetivo de alcanzar una meta.

Para esto inicialmente se presenta el diseño de la señalización del entorno basándonos en cómo adquiere el robot la información y qué tipo de información puede percibir para alcanzar el objetivo, posteriormente mostraremos la estructura o clasificación del espacio donde se moverá el robot y finalmente el sistema de búsqueda del destino.

## 3.2. Señalización del entorno

Cuando nos movemos por entornos desconocidos necesitamos saber nuestra ubicación en el contexto espacial ( edificio, calle, barrio, ciudad...), la ubicación del lugar destino (o al menos la dirección en que se encuentra) y cómo ir a ese lugar. Al no tener conocimientos previos, la señalización del entorno se convierte en un elemento primordial para alcanzar el objetivo.

El origen de la señalización es tan antiguo como el hombre, el humano ideó procedimientos y herramientas que le sirvieran de soporte a su memoria y a su capacidad retentiva, sabiendo que ese conocimiento también debía heredarse a generaciones venideras. La importancia de la señalización está relacionado con la supervivencia, las señales marcaban rutas cuando los caminos no existían. Según [Cos, 1987] , señalar es un impulso “intuitivo” que tiene como principio fundamental saber desplazarse en un entorno hacia un lugar determinado, así como identificar lo objetos.

La señalización entonces, la podemos definir como el conjunto de elementos colocados en el entorno que, mediante indicaciones gráficas, auditivas o táctiles, guían y ayudan a las personas a navegar en un área o edificio. La señalización sirve de soporte para la orientación en un espacio a un lugar determinado y facilita la accesibilidad a los servicios requeridos dando seguridad en los desplazamientos y las acciones.

### 3.2.1. El signo

La palabra signo proviene del término latino *signum*. Según la RAE <sup>1</sup>, un signo se trata de un objeto, fenómeno o acción material que, por naturaleza o por convención, representa o sustituye a otro. En [Avi, 2005], la autora define *el signo* como un *estímulo capaz de provocar una conducta en el organismo, particularmente en el sistema sensorial, el cual tiene funciones de abstracción, ordenamiento y recordatorio, para diferenciar cada significado*.

Si hablamos de sistemas de señalización, un signo es aquel elemento gráfico, auditivo o táctil usado para representar un lugar, objeto, entidad, una información o recomendación. Según la *Semiología* <sup>2</sup>, los elementos esenciales de un proceso semiótico son 4:

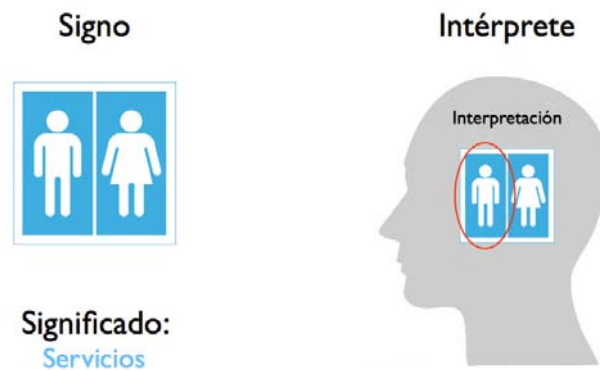
1. Signo.
2. Significado.
3. Intérprete.
4. Interpretación.

El *signo* se encarga de representar un mensaje (*significado*) que un receptor o *intérprete* debe decodificar para entenderlo o *interpretarlo* y pueda actuar de acuerdo al mensaje recibido. Esto es lo que se denomina un *proceso semiótico*.

<sup>1</sup> Real Academia Española

<sup>2</sup> “Ciencia que estudia la vida de los signos en la sociedad” Ferdinand de Saussure, 1908

La figura 3.1, muestra los diferentes elementos de un proceso semiótico, en este ejemplo en particular, el *signo* está representado por dos siluetas de un hombre y una mujer separados por una barra vertical, el *significado* es “Servicios”. El *intérprete* es la persona que ve el signo y desea acceder a los servicios. La *interpretación*, puede ser “puedo acceder por esa puerta a los servicios”.



**Fig. 3.1:** Elementos que intervienen en un proceso semiótico.

En [Conde, 2000], el autor resume la clasificación de los signos dependiendo de su naturaleza:

1. **Signos Naturales:** Son aquellos signos en que el hombre no ha participado en su creación. También son llamados indicios o indicaciones naturales. Tienen relación causa-efecto, por ejemplo: si hay humo, esto indica que hay fuego o si hay nubes negras es indicio de tormenta.
2. **Signos Artificiales:** Son signos creados por el hombre para un fin determinado. Los signos artificiales se dividen a su vez en:
  - a) Signos Lingüísticos: Son aquellos signos que incluyen sistemas verbales, bien sea hablados o escritos.
  - b) Signos No Lingüísticos: Estos signos no incluyen sistemas verbales. En este grupo están los símbolos e iconos.

### 3.2.2. La señal

En el contexto de un sistema de señalización para navegación podemos definir *señal* como un tipo de signo colocado en el entorno para identificar un elemento o lugar, o bien para indicar un aviso, dar una advertencia o una indicación. La señal además de identificar, sirve de guía para orientar a un individuo en el entorno.

### Clasificación de las señales

Las señales dependen de la función que realizan y del tipo de información que proporcionen al usuario [Fol, 1979] , estas pueden ser:

**Señales Orientativas** El objetivo de estas señales es ubicar al usuario respecto a su entorno, generalmente son representadas mediante un plano o mapa (Fig. 3.2).



**Fig. 3.2:** Mapa de la Rosaleda Cecilio Rodriguez, Parque de El Retiro - Madrid, con la señalización *Usted está aquí*.

**Señales Informativas** Son las que proporcionan una información específica de interés, por ejemplo: horarios y servicios, o los directorios que se colocan en la entrada de un edificio público (Fig. 3.3).

**Señales Identificativas** Son las que identifican un lugar específico. También son denominadas indicativas (Fig. 3.4).

**Señales Direccionales** Son signos que indican la ruta a seguir para llegar a un destino determinado. Generalmente son representadas mediante flechas. Se colocan en puntos estratégicos donde se debe escoger un camino (Fig. 3.5).

**Señales de Advertencia** Como su nombre lo indica, proporcionan anticipadamente un advertencia sobre alguna característica particular en el entorno que pueda afectar la navegación. Estas pueden ser preventivas o prohibitivas (Fig. 3.6).



**Fig. 3.3:** Ejemplo de señal informativa: Horario de trenes en la estación de Atocha (Madrid).



**Fig. 3.4:** Ejemplo de señales identificativas: Hotel Santander y el Teatro Reina Victoria - Madrid.





**Fig. 3.5:** Ejemplo de señales direccionales: Señal ubicada cerca de la Universidad Carlos III.



**Fig. 3.6:** Ejemplo de señal de advertencia: Atención, suelo mojado.



### 3.2.3. Señalización en Edificios

Los urbanistas utilizan diferentes recursos que pueden ser arquitectónicos, gráficos, auditivos o táctiles para ayudar a las personas a llegar al destino [Muhlhausen, 2000, Onc, 2009].

En relación a los elementos arquitectónicos los urbanistas generalmente siguen las siguientes pautas:

- Los espacios deben ser adecuados para la transitabilidad.
- Las señales y/o hitos memorables se deben situar a lo largo de los pasillos y en los principales puntos de decisión/intersección.
- Los puntos de información pública del edificio deben ser visibles desde la puerta de acceso.
- Los rótulos y elementos direccionales deben estar en armonía con el edificio y/o espacios.

Con respecto a los recursos gráficos, existen diversos criterios y recomendaciones:

- Los nombres de edificios, servicios y destinos deben estar normalizados y ser mostrados de forma coherente.
- La legibilidad y visibilidad debe ser adecuada.
- Proporcionar la información de la ubicación actual (p.ej. “Usted está aquí”).
- Colocar puntos de información en las zonas de salida, entrada y en los principales puntos de decisión o intersección.

En España, las pautas mencionadas siguen las normas o requisitos de accesibilidad DALCO (Deambulaci3n, Aprehensi3n, Localizaci3n y COmunicaci3n) [IEN, 2007]:

- Deambulaci3n: Se refiere a la movilidad y al estudio de los accesos en un edificio o entorno, p.ej. accesos, anchos de paso, anchos de giro, obst3culos, desniveles verticales (rampas, escaleras, ascensores).
- Aprehensi3n: Se refiere no s3lo al alcance f3sico (p.ej. interruptores y manillas), sino tambi3n el alcance auditivo (p.ej. avisos de megafon3a) y el visual (se3ales visuales).
- Localizaci3n: El usuario debe conocer en cada momento donde se encuentra y donde puede encontrar informaci3n para llegar a su destino.
- Comunicaci3n: Incluye la se3alizaci3n y los sistemas de aviso, alarmas, megafon3a, planos, etc.

Así como para las personas existen recomendaciones de diseño e implementación de recursos para navegación en el entorno, en un robot móvil es necesario adaptar estas reglas dependiendo de la morfología del robot y de su sistema sensorial. El robot puede obtener información de las personas usando reconocimiento de voz, visión o la información que pueda percibir con sensores como antenas RFID.

El sistema que se presenta en esta tesis, parte del requerimiento inicial de que el robot se moverá en entornos interiores y estructurados, por lo que la accesibilidad arquitectónica ya está dada por la estructura, por lo que entonces, es necesario diseñar y estructurar la señalización e información que el robot debe adquirir del entorno para llegar satisfactoriamente a la meta establecida. Esto implica diseñar una representación del conocimiento del entorno para modelar los lugares, acciones y señales que el robot usará para guiarse hacia la meta, situarse y moverse en el contexto espacial.

### 3.3. Diseño del sistema de señalización para navegación con robots móviles

El objetivo de un sistema de señalización es orientar e informar a los usuarios con la finalidad de que puedan llegar a su destino. El diseño de un sistema de signos depende de las necesidades de orientación de los usuarios, en este trabajo los usuarios son robots móviles. Un robot debe ser capaz de acceder a la información fácil y eficazmente para navegar en el entorno sin perderse y sin afectar la navegación de otros individuos en el entorno.

#### 3.3.1. Diseño de las señales

Las señales que se coloquen en el entorno son pre-diseñadas dependiendo del problema que pueda surgir al navegante en el entorno. Para ésto se debe tomar en cuenta los siguientes aspectos:

- La información contenida en las señales debe facilitar a través de un **mensaje claro** el conocimiento que el robot necesita del entorno.
- La **calidad de la información** también es importante, no debe ser confusa y tampoco debe ser excesiva o demasiado extensa.
- La **ubicación de las señales** debe ser accesible en el momento oportuno. En los robots (al igual que en los humanos), dependerá del sensor que recibe la información, por ejemplo: si es gráfica a una visibilidad adecuada, si es sonora a una distancia que sea audible.
- Las señales **no deben afectar la armonía del entorno**, es decir, no debe afectar estéticamente el entorno ni obstruir o dificultar la navegación.

### ¿Qué tipo de información es necesaria?

La información que se use del entorno dependerá del sistema sensorial del robot, de la estructura del entorno y de la información que el robot requiera para ejecutar las acciones que le permitan llegar al destino. Si no se tiene un conocimiento previo del entorno y se desea llegar a un destino determinado, la información de las señales deberá suministrar la respuesta a estas 3 preguntas claves: ¿dónde estoy?, ¿cómo voy? y ¿hacia dónde puedo ir?.

#### 1. ¿Dónde estoy?

Uno de los principales problemas que debe resolver un robot para su autonomía es la *localización*. Es necesario entonces, diseñar **Señales de Contenido o Identificativas**, que indique el nombre del lugar donde se encuentre el robot o alguna característica que lo identifique, por ejemplo, si es una oficina en un edificio público, especificar su función o el servicio que presta. La información que indican estas señales puede ser de dos tipos:

- a) **Información Geométrica:** Corresponde a las coordenadas  $(x, y, \theta)$  del robot en relación a un sistema de referencia. La localización geométrica es necesaria cuando la información topológica es insuficiente o si nos encontramos en entornos no estructurados, por ejemplo, cuando se está dentro de una sala de estar o de un laboratorio con sólo mesas y sillas. En estos casos es necesario tener un mapa geométrico del entorno para que el robot pueda hacer movimientos precisos, como ir a un punto geométrico concreto. En esta tesis, la localización la obtenemos usando el algoritmo *AMCL* (*Adaptive Monte-Carlo Localization*) [Fox et al., 1999] implementado en el proyecto *Player* [Gerkey et al., 2003].
- b) **Información Topológica:** La localización topológica ubica al usuario en relación a los elementos representativos del entorno. Por ejemplo: “*Estas en el Laboratorio de Robótica frente al despacho de la directora del departamento*”. Es un tipo de información basada en el *lenguaje natural*. Se refiere a la identidad semántica de los elementos del entorno y su relación. En este trabajo se divide en 2 tipos:
  - 1) *Conceptos*: Refiriéndose a la entidad de cada lugar o parte del lugar (zonas o regiones), sus propiedades, funcionalidad y si pertenece a una categoría específica. Por ejemplo, si estamos en un edificio de la universidad los posibles conceptos son: despachos (ej. despacho del director), aulas (ej. aula informática), laboratorio (ej. laboratorio de robótica), oficinas, aseos, etc.
  - 2) *Relaciones*: Especifica las relaciones espaciales entre conceptos. Por ejemplo, antes de, después de, al lado de, a la derecha de, a la izquierda de, adelante y atrás.

La representación del entorno a través de conceptos es una manera de simplificar el conocimiento que tenemos del entorno. Con este objetivo modelamos

una especificación del conocimiento del entorno basándonos en los siguientes conceptos fundamentales:

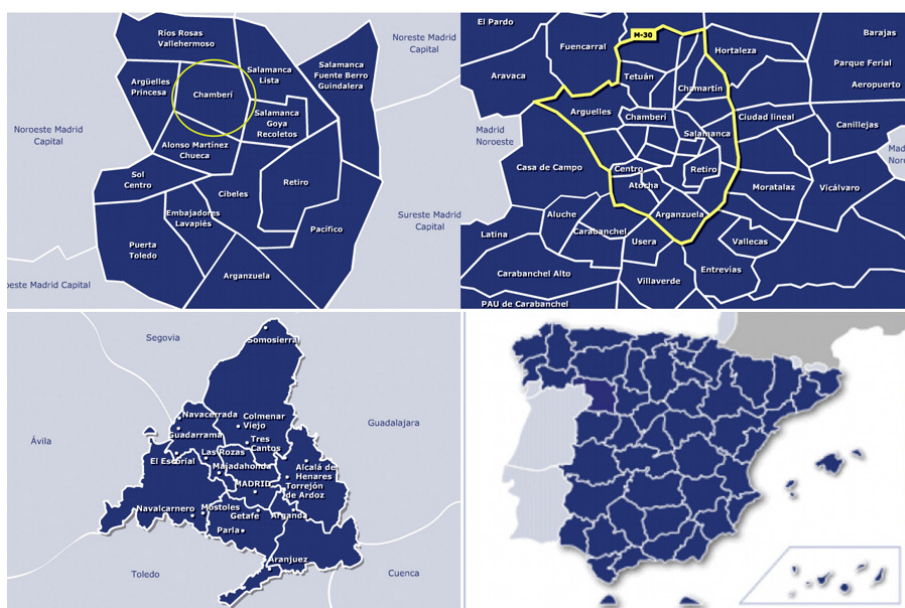
**Zona** Geográficamente hablando, una zona designa un área o extensión determinada de tierra más grande que las subzonas o subdivisiones que la constituye. En este trabajo cada zona posee un nombre y un identificador alfa-numérico.

**Lugar** Según la Real Academia Española, *lugar* se refiere a un “*Espacio ocupado o que puede ser ocupado por un cuerpo cualquiera. Sitio o paraje.*” Un lugar es la representación explícita de un espacio físico, sitio o entidad donde un individuo puede estar. En este trabajo, un **lugar** está definido por la representación jerárquica de las zonas anidadas que lo contienen.

Estas definiciones parten del modelo mental que tenemos las personas para definir un sitio. Para ilustrar mejor los dos conceptos anteriores, veamos el siguiente ejemplo, supongamos que queremos definir el lugar “Barrio Chamberí” ubicado en la ciudad de Madrid (España), el *lugar* entonces será definido por las siguientes *zonas*:

*Barrio Chamberí → Distrito Madrid Centro → Madrid Capital →  
Comunidad de Madrid → España*

En la figura 3.7, se muestran los mapas donde se muestra la ubicación geográfica del Barrio Chamberí, siguiendo la secuencia de zonas anidadas <sup>3</sup>.



**Fig. 3.7:** Mapas de las zonas geográficas para definir el lugar Barrio de Chamberí.

<sup>3</sup> Mapas tomados de la web inmobiliaria [www.fotocasa.es](http://www.fotocasa.es)

La mayoría de los edificios públicos usan la clasificación por zonas, por ejemplo en los aeropuertos, si tienes que viajar hacia un destino específico, es necesario saber en qué terminal esta la aerolínea, luego buscar en qué zona y planta del edificio se encuentra. Una vez que hayas facturado debes buscar la puerta de embarque que será identificada por una zona y número de puerta determinada. En la figura 3.8 se muestra el mapa del aeropuerto de Madrid <sup>4</sup> con sus correspondientes zonas divididas en las terminales T1, T2, T3, T4 y T4S.



**Fig. 3.8:** Mapa del Aeropuerto Internacional de Barajas en Madrid.

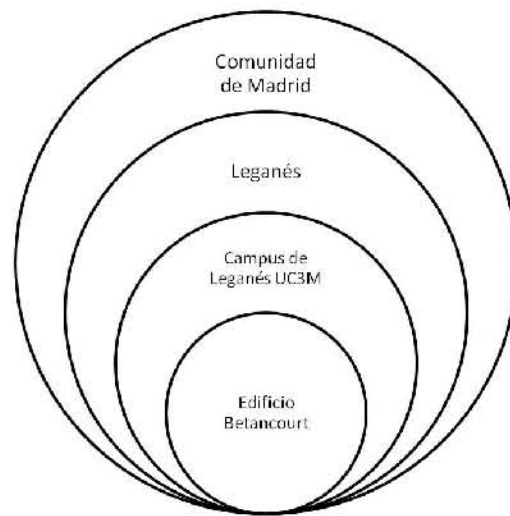
Si queremos definir el *lugar* Edificio Betancourt de la Universidad Carlos III de Madrid, la representación general de este lugar estaría conformado por las siguientes *zonas* (ordenado desde lo más específico a lo más genérico) (Fig. 3.9):

- Edificio Betancourt.
- Campus de Leganés de la UC3M.
- Leganés (Población).
- Madrid (Comunidad).

Para facilitar la identificación, búsqueda y relación entre lugares, cada zona posee un identificador alfanumérico, esto es una regla que generalmente se usa en la señalización de edificios. Así por ejemplo en el Campus de Leganés de la UC3M, cada edificio posee un número de identificación, tal y como muestra la figura 3.10.

Así mismo, en cada edificio del campus hay un mapa que especifica la ubicación e identificación de las zonas que la conforman, cada zona posee una letra que la identifica (Fig. 3.11).

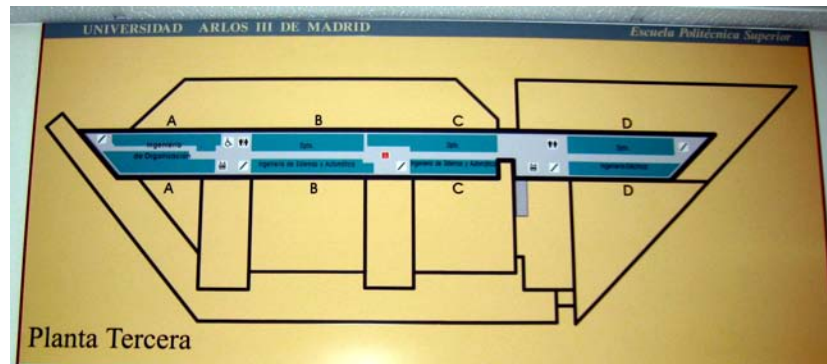
<sup>4</sup> Aeropuertos Españoles y Navegación Aérea <http://www.aena.es>



**Fig. 3.9:** Representación del lugar Edificio Betancourt de la UC3M y el conjunto de zonas anidadas al que pertenece.

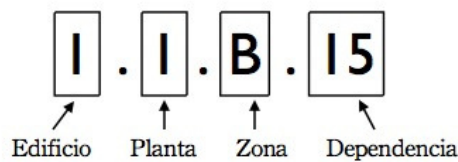


**Fig. 3.10:** Plano de la Escuela Politécnica Superior. Campus de Leganés UC3M.



**Fig. 3.11:** Definición del lugar: Mapa de la tercera planta del Edificio Betancourt de la UC3M.

Las diversas zonas de cada edificio están identificadas por las letras de la A a la J. Todos los despachos, salas y dependencias están identificados por un código alfanumérico. Cada lugar del edificio utiliza el esquema mostrado en la figura 3.12.



**Fig. 3.12:** Código alfanumérico para la identificación de despachos y salas en la UC3M.

## 2. ¿Cómo voy? y 3. ¿Hacia dónde puedo ir?

Como se ha dicho anteriormente, uno de los objetivos de un sistema de señalización consiste en determinar cursos de acción para que un usuario pueda moverse en el entorno y facilitar su llegada a la meta. El tipo de señales usadas para lograr este propósito son las **Señales Direccionales**. Estas señales están compuestas por dos tipos de información: una dirección y un lugar. Este tipo de señales las encontramos frecuentemente en el entorno, por ejemplo, si estamos en el aeropuerto y debemos dirigirnos a un mostrador específico, siguiendo señales como la mostrada en la figura 3.13, llegaremos a nuestro destino satisfactoriamente.

Para el sistema de señalización de esta tesis definiremos dos conceptos que indicarán al robot cómo y hacia dónde ir:

**Conexión** Un *lugar* puede estar *conectado* directa o indirectamente a otro *lugar*, una **conexión** entonces se define como el posible lugar o lugares a donde puede ir el robot desde el lugar donde se encuentra.

Una conexión está conformada por un *lugar* o *lista de lugares* y una *acción*.



**Fig. 3.13:** Señalización en el Aeropuerto Internacional de Barajas en Madrid.

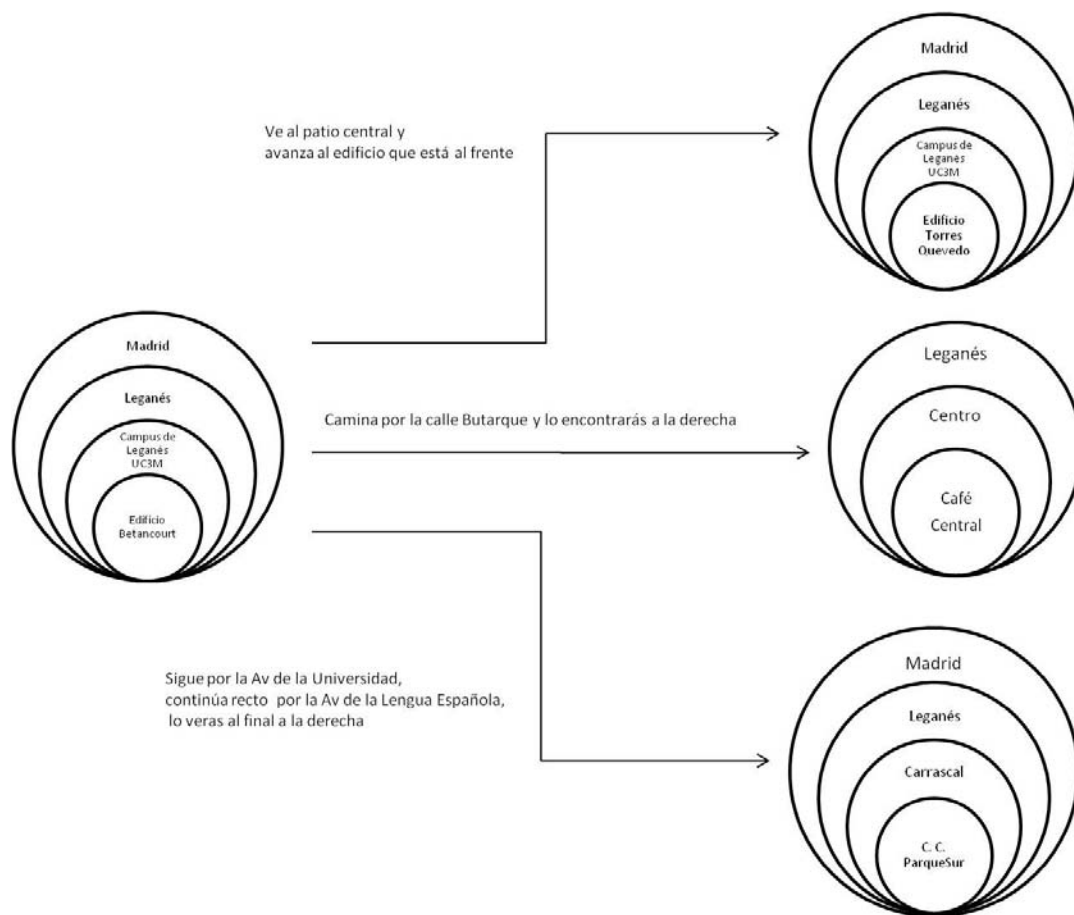
**Acción** Indica la acción que el robot debe ejecutar para llegar al lugar que indica la conexión. La acción puede ser:

- Topológica: por ejemplo: “sigue por la derecha”, “avanza”, “sigue el contorno por la izquierda”, etc.
- Geométrica: si se tiene un mapa geométrico del entorno la indicación puede señalar al robot ir a una coordenada específica  $(x, y, \theta)$ .

En la figura 3.14, se muestra el lugar: Edificio Butarque de la UC3M y 3 posibles conexiones a donde se puede llegar siguiendo las indicaciones (*acciones*):

- Conexión 1:  
Lugar: *Edificio Torres Quevedo* → *Campus de Leganés UC3M* → *Leganés* → *Campus de Leganés UC3M* → *Madrid*  
Acción: *Ve al patio central y avanza al edificio que está al frente.*
- Conexión 2:  
Lugar: *Café Central* → *Centro* → *Leganés*  
Acción: *Camina por la calle Butarque y lo encontrarás a la derecha.*
- Conexión 3: Lugar: *C. C. ParqueSur* → *Carrascal* → *Leganés* → *Madrid*  
Acción: *Sigue por la Av de la Universidad, continúa recto por la Av de la Lengua Española, lo veras al final a la izquierda.*





**Fig. 3.14:** Edificio Betancourt de la UC3M, y algunas conexiones.

### 3.3.2. Estructura de la información

Para almacenar la información del entorno se debe considerar no sólo su contenido sino también su estructura, ya que el conocimiento del entorno debe ser procesado y no únicamente representado. Por esa razón es necesario utilizar herramientas que permitan modelar la estructura de manera fácil, estructurada y eficiente (Ver más en 6.4) .

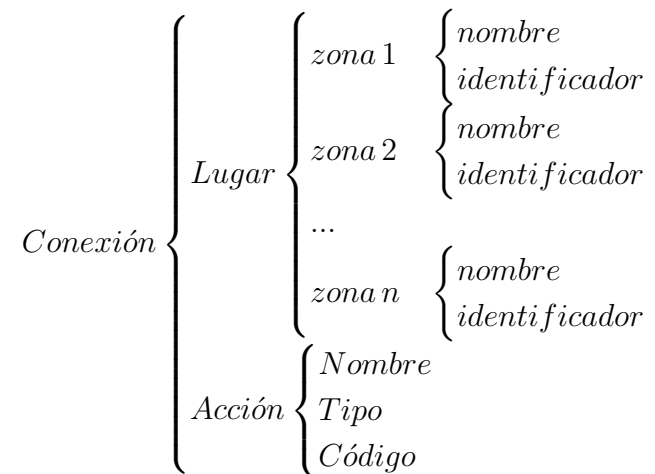
La estructura básica del entorno la modelamos de la siguiente manera:

1. **Lugar:** Un *lugar* está conformado por *zonas*, cada zona posee dos elementos: un *nombre* y un *identificador* alfanumérico.

$$\text{Lugar} \left\{ \begin{array}{l} \text{zona 1} \left\{ \begin{array}{l} \text{nombre} \\ \text{identificador} \end{array} \right. \\ \text{zona 2} \left\{ \begin{array}{l} \text{nombre} \\ \text{identificador} \end{array} \right. \\ \dots \\ \text{zona } n \left\{ \begin{array}{l} \text{nombre} \\ \text{identificador} \end{array} \right. \end{array} \right.$$

2. **Conexiones:** Cada conexión está conformada por un *lugar* y una *acción* (o lista de acciones) que se deben realizar para llegar al lugar indicado. No necesariamente este lugar debe estar directamente conectado (físicamente) al lugar. Cada *acción* posee 3 elementos adicionales:

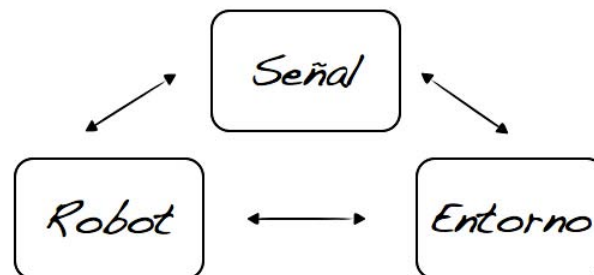
- Nombre: corresponde a la definición de la acción que se debe realizar.
- Tipo: identifica el tipo de acción, si es topológica (p.ej. “sigue el contorno a la derecha”), o geométrica (“ve a la coordenada  $(x, y, \theta)$ ”). En el caso de ser geométrica se debe especificar una etiqueta adicional en el fichero XML que contenga la dirección donde está almacenado el mapa.
- Código: es una etiqueta adicional que usa el sistema de navegación del robot que dependerá de la habilidad que debe activarse para ejecutar la acción.



### 3.3.3. Diseño físico de las señales. Almacenamiento de la información

En un sistema de señalización existen 3 factores importantes relacionados entre sí que influyen en el momento de realizar un diseño (Fig. 3.15):

- El usuario del sistema de señalización, es decir, el robot.
- El entorno donde se colocarán las señales.
- La señal física en sí.



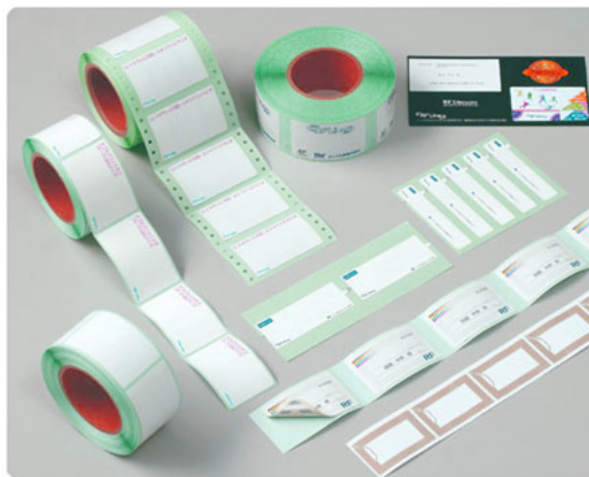
**Fig. 3.15:** Factores que intervienen en el diseño de un sistema de señalización.

El diseño debe cuidar la armonía entre estos 3 elementos, por eso es importante considerar los siguientes aspectos:

1. **Armonía con el entorno.** Las señales deben estar en armonía con el entorno y no obstruir la circulación de las personas ni la del robot, además de ser compatibles con las dimensiones, necesidades y habilidades de los usuarios del sistema (robots móviles).

2. **El contenido.** Este aspecto es importante tanto para el robot como para la señal. La información debe ser eficiente y concisa para que el robot la pueda interpretar y posteriormente realizar las tareas necesarias para llegar a la meta. En cuanto a la señal, debe ser diseñada para que pueda almacenar físicamente dicha información.
3. **Tipo de información que las señales proporcionarán.** Ésto está directamente relacionado con el sensor que el robot usará para adquirir la información, es decir, puede ser visual, auditiva, o la que pueda ser percibida a través de señales RFID.
4. **Correcta ubicación de las señales.** Ésto implica que sean accesibles al robot, es decir que estén dentro del rango de lectura de su sistema sensorial. La ubicación de las señales también debe ser sistemática, que haga previsible su localización y evitar la aglomeración de señales que traiga como consecuencia que el robot pueda confundirse y no llegar a la meta.

Siguiendo el modelo de información de las señales explicadas en la sección anterior y los factores que intervienen en el diseño de un sistema de señalización, en esta tesis hemos optado por almacenar la información en etiquetas pasivas RFID (Fig. 3.16), ya que cumplen con las premisas de señalización expuestas previamente:



**Fig. 3.16:** Etiquetas RFID pasivas. (Fotografía de <http://www.rfideas4u.com/>)

1. **Armonía en el entorno:** Las etiquetas RFID se caracterizan por su flexibilidad, delgadez y reducido tamaño. Algunas de las etiquetas son adhesivas, por lo que resulta fácil colocarlas en el entorno sin que obstaculicen la navegación ni de las personas ni de los robots. Además, pueden pasar desapercibidas para las personas si, por ejemplo, se pintan del mismo color de la superficie donde se encuentren.

2. **Contenido:** El contenido de las etiquetas RFID es fácil de crear y modificar. La información la puede almacenar una persona o un robot. Adicionalmente, las etiquetas RFID poseen una adecuada memoria de almacenamiento de acuerdo a la información que se ha diseñado como una señal. Las etiquetas RFID almacenarán la siguiente información:
  - UID: Es un identificador único de la etiqueta asignado por el fabricante.
  - Un identificador que representa el tipo de información que guarda la etiqueta: información para navegación o información útil para reconocimiento de objetos.
  - La información de la señal: lugar, conexiones y acciones.
3. **Ubicación de las señales:** Estas etiquetas serán colocadas en el entorno en puntos clave del entorno (como esquinas, cruces, entradas, etc.), donde el robot tenga acceso para leerlas e incluso guardar información en éstas. Ésto se detallará más adelante en el capítulo 6.
4. **Coste:** Una de las ventajas adicionales de usar la tecnología RFID es el bajo coste de señalizar el entorno( Cada etiqueta puede tener un coste aproximado de 20 céntimos de euro).

### 3.4. Discusión

Según Craig Berger [Berger, 2009], el término *Wayfinding*, en su definición más simple, es el acto de encontrar el camino a un destino específico. El objetivo de diseñar un sistema de *Wayfinding* (sistema de señalización), consiste en ayudar a las personas a encontrar el camino para llegar a su destino.

En el capítulo presentado, se exponen las bases iniciales para el diseño de un sistema de señalización para la navegación de un robot móvil. El objetivo es el mismo que un sistema de *Wayfinding*, solo que aplicado a los robots: *ayudar a los robots a encontrar el camino para llegar a su destino*. El diseño realizado parte de los fundamentos de los sistemas de señalización para las personas, desde el estudio de los diferentes tipos de señales que encontramos en el entorno hasta las diferentes reglas y estándares establecidas para la correcta señalización del entorno. Basándonos en ésto, se ha propuesto el diseño de una *señal* que un robot pueda leer y le sirva de ayuda en su misión de navegación. Este diseño incluye: el diseño lógico y el diseño físico. Para el diseño lógico se han definido cuatro conceptos clave en el sistema de señalización: *zona*, *lugar conexión y acción* y para el diseño físico se ha escogido usar etiquetas RFID por cumplir con características adecuadas de armonía y almacenamiento, además que es una tecnología adecuada para incorporar en un robot.



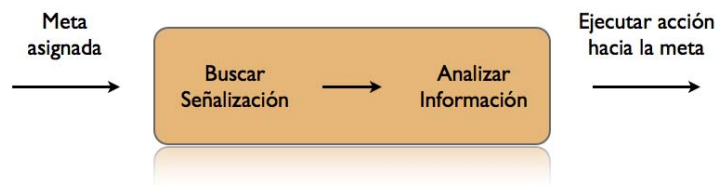
## 4. SISTEMA DE NAVEGACIÓN MEDIANTE SEÑALES

### 4.1. Introducción

El objetivo principal del sistema de navegación que se presenta en este capítulo, es lograr la implementación de un proceso de búsqueda y orientación del robot en un entorno desconocido, usando elementos del entorno (específicamente la señalización), con la finalidad de llegar a una meta deseada. El algoritmo desarrollado partirá de las siguiente premisas:

1. El robot tiene una meta asignada.
2. El entorno es desconocido.
3. El robot no memoriza los lugares que ha visitado.
4. Para llegar a la meta, el sistema usará la información del sistema de señalización explicado en el capítulo 3.

El proceso de orientación y búsqueda de la meta en un robot, es básicamente similar al que hacemos los humanos cuando necesitamos llegar a un destino y desconocemos el camino y el entorno. Inicialmente tenemos una meta asignada, posteriormente buscamos un cartel o señal que nos oriente hacia el destino (búsqueda de la señalización), seguidamente, una vez obtenida la información, la analizamos y realizamos la acción que nos ayude a llegar a la meta (Fig. 4.1).



**Fig. 4.1:** Proceso de orientación y búsqueda de la meta en entornos desconocidos.

## 4.2. Definiciones

Antes de desarrollar el algoritmo de navegación, definiremos los elementos que intervienen en el mismo.

**Zona** Se define por **zona**  $z$ , a una región o área particular del entorno  $E$ .

Cada zona  $z$ , posee un nombre y un identificador alfa-numérico.

**Lugar** Se define por **lugar**  $l_i$ , a la representación única y explícita de un espacio físico, sitio o entidad donde un robot puede estar. Está constituido por una o varias zonas anidadas.

$$l_i = \{z_1, \dots, z_n\} \quad (4.1)$$

Donde  $n$  es el número de zonas que conforman el lugar  $l_i$ .

Si existen varias zonas que conforman un lugar, estas son anidadas, como se muestra en 4.2.

$$z_1 \subseteq z_2 \dots \subseteq z_n \quad (4.2)$$

Así por ejemplo, cuando hablamos del “Despacho 1.3.B.15” de la UC3M, en su representación más simple, el lugar estaría conformado por las siguientes zonas:

$$l = \{z_1, z_2, z_3, z_4\} \quad (4.3)$$

$z_1 = \text{Despacho 1.3.B.15}$

$z_2 = \text{Zona B 1.3.B}$

$z_3 = \text{Tercera Planta 1.3}$

$z_4 = \text{Edif. Betancourt 1}$

Para un lugar  $l_i$  conformado por las zonas  $\{z_1, \dots, z_n\}$  que cumplen con la ecuación 4.2, entonces, la zona más específica o lugar concreto  $z_n$ , posee un identificador dado por un carácter alfanumérico, al que llamaremos  $id_n$ , la siguiente zona  $z_{n+1}$ , tendrá un identificador con un nivel de abstracción inferior al de  $z_n$ . Entonces, para el ejemplo anterior los identificadores pueden ser:

$id_1 = 1.3.B.15$

$id_2 = 1.3.B$

$id_3 = 1.3$

$id_4 = 1$

Esto lo hemos ilustrado mejor, en el ejemplo visto en la figura 3.12, donde vemos que en la Universidad Carlos III de Madrid, todos los despachos, salas y dependencias están identificados por un código alfanumérico. Este tipo de estructura es lo que se usa generalmente en la señalización de los edificios públicos y es lo que seguimos en esta tesis. Cabe destacar que la representación del ejemplo



anterior es simple, las zonas genéricas pueden extenderse desde el Edificio Betancurt de la UC3M a zonas como: Campus de Leganés, Leganés, Comunidad de Madrid, España, Europa...

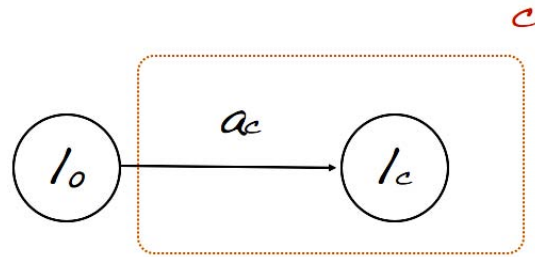
**Conexión** Una **conexión**  $c_i$  indica el lugar  $l_c$  a donde puede ir un robot desde el lugar indicado por la señal  $l_0$ , ejecutando una **acción**  $a_c$  (Fig. 4.2).

Una conexión está representada por  $((l_c, id_c)a_c)$ , donde:

$l_c$  = Nombre del lugar que indica la conexión.

$id_c$  = Identificador del lugar  $l_c$ .

$a_c$  = Acción que se debe realizar para llegar al lugar que indica la conexión.



**Fig. 4.2:** Representación de un lugar  $l_0$  y una conexión  $c_i$

**Señal** Una señal  $s_0$  está conformado por un lugar  $l_0$  que indica el lugar actual y una conexión  $c_i$  o lista de conexiones  $c_i, \dots, c_n$ , que indican al robot los posibles lugares a donde puede ir (Fig. 4.3).

Físicamente, las señales dependerán del sensor que sea utilizado por el robot, como se ha explicado en la sección 3.3.3, las señales que se usarán para esta tesis son etiquetas RFID pasivas, en las que se almacenará la información semiótica de las señales.

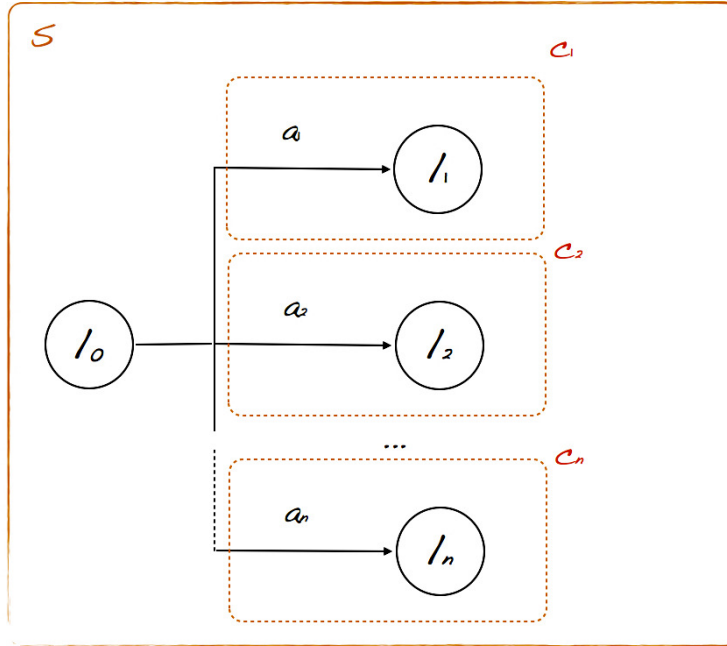
### 4.3. Algoritmo de Navegación

Las etiquetas RFID, que se usarán como señales para el sistema de navegación, estarán colocadas en el entorno y accesibles al robot durante su navegación. De este modo, el proceso partirá de una primera fase de búsqueda de señales que denominaremos exploración o búsqueda de etiquetas RFID.

Se parte inicialmente con el supuesto de que el robot tiene una meta asignada. El robot comenzará a explorar el entorno buscando las señales colocadas en el entorno (etiquetas RFID), al encontrar una señal procede a leerla y analizar los datos para ejecutar las acciones necesarias para llegar a la meta.

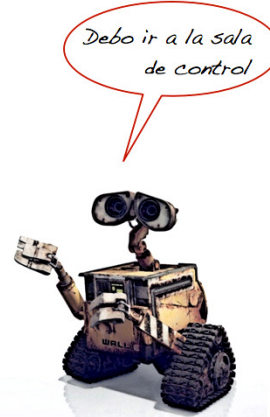
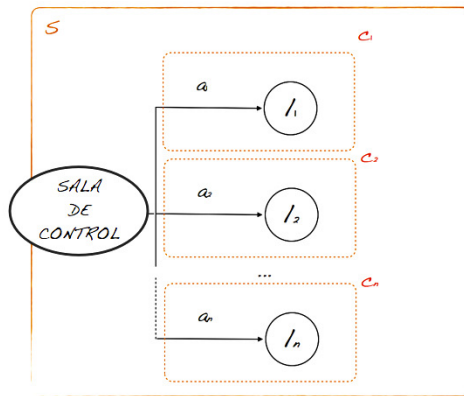
Así pues, tendremos los siguientes casos:

**Caso 1: La meta coincide con el lugar actual.** Es el caso más básico. El robot encuentra una señal ( $s_i$ ), lee la señal y la interpreta, compara el nombre y/o el identificador de la meta asignada ( $l_M$ ) con el lugar actual que indica la etiqueta ( $l_0$ ).



**Fig. 4.3:** Representación de una señal.

En este caso  $l_M = l_0 \Rightarrow$  el robot ha llegado a la meta y el objetivo es alcanzado (Fig. 4.4).



**Fig. 4.4:** Caso 1: La meta coincide con el lugar que señala la etiqueta.

Caso 2: La meta está señalizada directamente en una conexión. Si  $l_M \neq l_0$ , entonces se debe comparar el lugar  $l_M$  con el lugar que indica cada conexión  $c_i$  de la señal

hacia donde el robot puede ir desde el lugar actual  $l_0$ .

En caso de existir una conexión que señale hacia la meta, el robot ejecuta la acción indicada y se dirige hacia la meta.

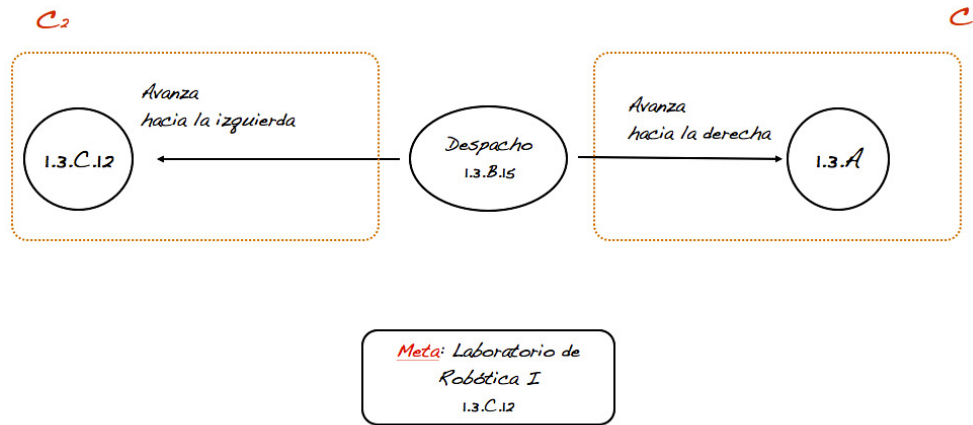
**Ejemplo** Si el lugar actual es el “*Despacho 1.3.B.15*”, y las conexiones están dadas por  $((l_c, id_c), a_c)$ :

$c_1 = \{(\text{Departamento de Ingeniería de Organización}, 1.3.A), \text{Avanza hacia la derecha}\}$

$c_2 = \{(\text{Laboratorio de Robótica I}, 1.3.C.12), \text{Avanza hacia la izquierda}\}$

y si la meta  $l_m = (\text{Laboratorio de Robótica I}, 1.3.C.12)$ ,

entonces el robot ejecutará la acción que indica  $c_2 = \text{Avanza hacia la izquierda}$ , ya que  $l_m = l_1$ . (Fig. 4.5)



**Fig. 4.5:** Caso 2: La meta coincide con una conexión.

**Caso 3:** La meta está señalizada indirectamente en una conexión. Si al comparar las conexiones la señal no indica explícitamente cómo llegar a la meta, usamos el identificador que tiene cada zona para orientar al robot hacia la meta.

**Ejemplo** Si el lugar actual es el “*Despacho 1.3.B.15*”, y las conexiones están dadas por  $((l_c, id_c), a_c)$ :

$c_1 = \{(\text{Departamento de Ingeniería de Organización}, 1.3.A), \text{Avanza hacia la derecha}\}$

$c_2 = \{(\text{Laboratorio de Robótica I}, 1.3.C.12), \text{Avanza hacia la izquierda}\}$

y la meta  $l_M = (\text{Sala de Reuniones}, 1.3.A.05)$ ,

entonces el robot al comparar la meta y no coincidir con en el lugar actual, ni con las conexiones, buscará un identificador similar, es decir buscará un identificador genérico que esté contenido en el identificador de la meta que busca, esto es:

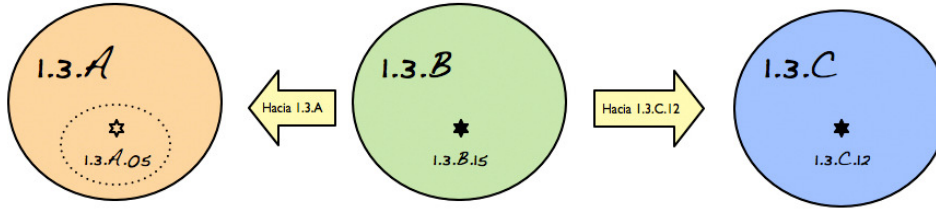
El identificador *1.3.A.05* lo descompone y realiza las comparaciones secuencialmente en el siguiente orden hasta que haya una coincidencia:

- *1.3.A.05*
- *1.3.A*

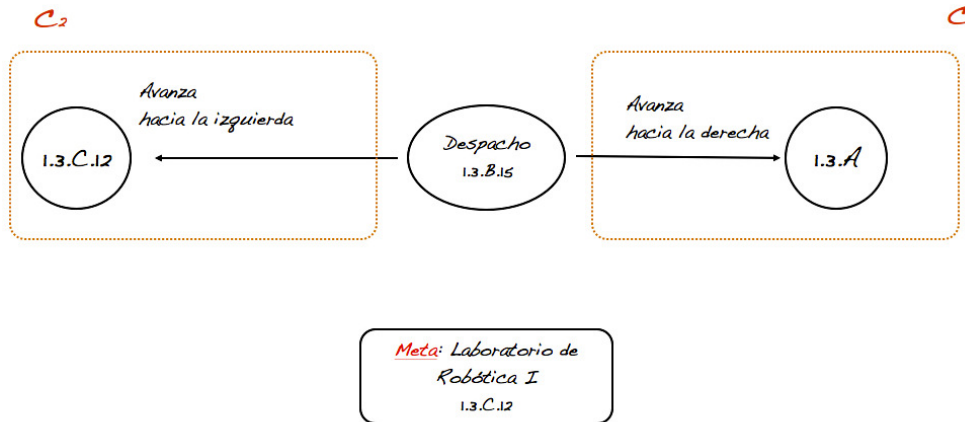
Compara entonces la zona *1.3.A*, si esta corresponde a una de las conexiones ejecuta la acción que indica la señal, si no es así, separa de nuevo hasta hacer todas las comparaciones de las zonas. Para el ejemplo, el robot encuentra la conexión a la zona *1.3.A* y ejecutará la acción que indica:

$c_1 = \text{Avanza hacia la derecha}$ ,

al llegar ahí encontrará de nuevo una señal y repetirá de nuevo el algoritmo hasta alcanzar la meta (Fig. 4.6 y Fig. 4.7).



**Fig. 4.6:** Señalización hacia una zona genérica



**Fig. 4.7:** Caso 3: La meta coincide con una zona de una conexión

Caso 4: La meta no está señalizada. Si no se cumple ninguno de los casos anteriores, el robot puede realizar alguna de las siguientes acciones:

- Explorar buscando una señal.

- Realizar la última acción que le haya indicado una señal que cumpliera con alguno de los casos 2 o 3.
- En espera a que se le asigne una nueva meta o en espera a recibir información externa (p.ej: a través de interacción humano-robot).

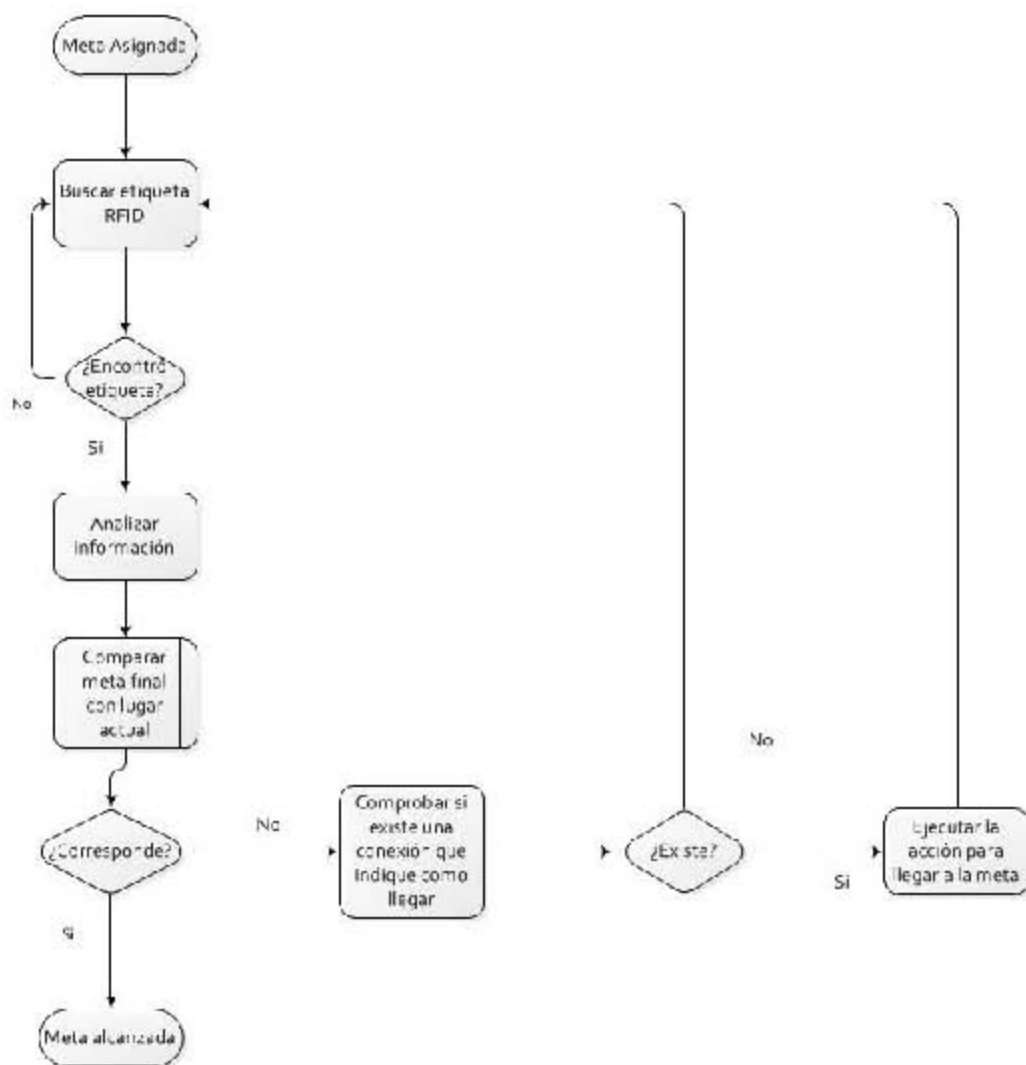
En este caso particularmente, en la vida real se usan señales genéricas que llevan a lugares comunes, como por ejemplo las señales que indican *Salida* o en las autopistas la señal de *Todas Direcciones*. Para este diseño pondremos conexiones a lugares genéricos para que el robot pueda tener otras opciones si se encuentra en el caso 4, así por ejemplo, si quiere moverse entre plantas del edificio, buscará la señal genérica *Ascensores*.



**Fig. 4.8:** Señal genérica indicando la salida

En resumen, el proceso desde que se obtiene la información contenida en la señal es el siguiente (Fig. 4.9):

1. Se compara la *meta asignada* con el *lugar* que indica la señal. Se compara el nombre del lugar y su identificador.
2. Si coinciden entonces significa que el robot ha llegado a la meta.
3. Si no coincide la meta con el lugar que señala la etiqueta, se busca si alguna conexión indica cómo ir a la meta. Esto es, comparar con los nombres y/o identificadores de las zonas que componen el lugar.
4. En caso de coincidir, se ejecuta la acción (o conjunto de acciones) que debe ejecutar el robot para alcanzar la meta o para alcanzar la señal próxima que lo lleve hacia ésta.
5. En caso de no coincidir, el robot sigue explorando para encontrar una nueva etiqueta que pueda orientarlo hacia la meta.



**Fig. 4.9:** Diagrama de actividades del algoritmo de navegación

## 4.4. Discusión

El algoritmo de navegación que se ha presentado, pretende ser una aproximación al modo en que las personas navegamos usando la señalización existente en el entorno. En este algoritmo el nivel de conocimiento que el robot tiene del entorno es nulo, es decir, nos encontramos en la tercera categoría de clasificación del problema de Wayfinding, planteado por Allen [Allen, 1999] *‘La meta corresponde a un destino desconocido’* o en la cuarta categoría según [Xia et al., 2008] *“Wayfinding cuando se desconoce el camino que se debe tomar pero existen puntos de referencia en el entorno”*. Este sistema de navegación, usa la información de un sistema de señalización colocado a priori en el entorno con el objetivo de guiar al robot para que alcance su objetivo de navegación. Los casos que hemos manejado son básicamente los que se presentan cuando nos encontramos en la situación de querer ir a un lugar sin conocimiento previo del entorno. Inicialmente buscamos una señal o aviso que nos proporcione una referencia de ¿dónde estamos? o ¿hacia dónde podemos ir?. Una vez que leemos la señal, interpretamos la información recibida y en base a ésta realizamos las acciones que nos aproximen a nuestro objetivo. Éstas acciones dependerán de los siguientes casos: la meta coincide con la ubicación de la señal (¿dónde estoy?), la meta coincide con una indicación (¿hacia dónde puedo ir?), la meta coincide parcialmente con los datos de una indicación o, el último caso, que no haya información en la señal sobre el destino. El objetivo del algoritmo es que el robot navegue y llegue a un lugar destino, sin tomar en cuenta la precisión. Aunque existen diferentes modelos matemáticos que aproximan el proceso de navegación, tomando en cuenta factores como precisión geométrica en los movimientos, nuestra aproximación pretende acercarse más a la forma natural de navegación de los humanos, donde los movimientos y acciones pueden ser inexactas ya que las personas no aprendemos métricas Euclidianas para la navegación [Freundschuh, 1991].





## 5. SEÑALIZACIÓN BASADA EN LA INTERACCIÓN HUMANO-ROBOT

### 5.1. Introducción

Cuando circulamos por una carretera, la señalización existente en la vía, nos permite tener una referencia sobre nuestra ubicación y el lugar hacia donde nos dirigimos. Estas señales están colocadas siguiendo un estándar, que en el caso de España, lo dicta la Dirección General de Tráfico (DGT)<sup>1</sup>. Este organismo se encarga de diseñar, escribir y colocar las señales de tráfico, según criterios establecidos en la legislación. Las señales permanecen fijas e inalterables hasta que haya una modificación importante en la carretera. Evidentemente, estas modificaciones sólo puede hacerlas el organismo competente, los usuarios no pueden alterar dichas señales.

Sin embargo, en la vida real existe otro tipo de sistemas que los usuarios pueden modificar, si bien no es un sistema de señalización, sí tienen que ver con la información, como las páginas Wiki. La Wikipedia, por ejemplo, según su lema se define como *“La enciclopedia libre que todos pueden editar”*, donde los usuarios pueden agregar y/o modificar información que pueda ser útil para otros usuarios en el futuro. La ventaja de las Wikis es que permite crear y mejorar las páginas de forma rápida, dando libertad al usuario de que participe en su edición, a diferencia de otros sistemas típicos, donde es más difícil que los usuarios contribuyan a enriquecer y mejorar la información.

La importancia de que el robot pueda señalar el entorno, (en este caso, almacenar información en las etiquetas RFID), está relacionada directamente con la automatización del sistema.

Supongamos que se tiene un robot en el hogar. La persona que adquiere el robot requerirá colocar el sistema de señalización para la navegación del robot. Para realizar esta tarea, el usuario probablemente no tendrá conocimientos técnicos para, por ejemplo, estructurar un fichero XML con el conjunto de elementos y relaciones del entorno que representan una señal; entonces en su lugar tendría que llamar un técnico calificado que instale el sistema y que lo actualice cada vez que el usuario requiera colocar nuevas señales o hacer una modificación en las mismas. Con la automatización de la escritura de las señales, una persona sin conocimientos técnicos previos, podrá comunicarse naturalmente con el robot e indicarle mediante interacción por voz, el sitio donde está y hacia donde puede ir.

Para la navegación de los robots en el contexto de esta tesis, en un principio,

---

<sup>1</sup> Dirección General de Tráfico [www.dgt.es](http://www.dgt.es)

las señales estarán previamente colocadas en el entorno, en puntos que se consideren clave para la navegación e identificación de lugares, como cruces, puertas, entrada principal, cambios de zonas, etc. Las etiquetas inicialmente pueden estar vacías o tener información de la señal. El robot al encontrar una señal (en este caso, una etiqueta RFID), podrá agregar y/o modificar la información contenida en la señal, con la finalidad de automatizar el proceso de señalización o de mejorar la información para uso propio en el futuro o para otros robots que naveguen en el entorno. Las señales entonces, serán modificadas por el robot con la información del modelo de una señal, explicado en los capítulos anteriores.

En este capítulo se describirá el proceso de señalización, cómo el robot adquiere la información para las señales, los posibles canales de información, cómo es el procesamiento y estructuración de los datos, el proceso de escritura de una señal y finalmente su implementación e integración en un sistema robótico.

## **5.2. Adquisición de la información**

Para iniciar el proceso de señalización, el robot debe adquirir la información que escribirá en la señal. Las fuentes de información pueden ser muy variadas, desde preguntar a una persona y que ésta le indique la información, hasta leerlo en otra señal y deducir lo que debe escribir. Esta tesis se enfoca principalmente en la adquisición de la información mediante la interacción humano-robot.

### **5.2.1. Adquisición de la información mediante interacción Humano-Robot**

Una de las metas fundamentales de la investigación en el campo de la robótica social es la interacción natural entre el hombre y el robot. En los humanos, al trasladarnos por sitios desconocidos, es una actitud cotidiana preguntar a otras personas el modo de llegar a un sitio específico si no disponemos de un mapa o no encontramos una señal que nos indique cómo llegar. Ya que los robots deben coexistir con los humanos al navegar en entornos estructurados e interiores, donde las personas son parte del entorno, como el hogar, hospitales, museos, oficinas, etc., es importante que se pueda compaginar las tareas de navegación del robot con la interacción con los humanos, de manera tal que se pueda complementar la información que el robot requiere del entorno, para alcanzar satisfactoriamente la meta y complete así la tarea de navegación.

Uno de los objetivos de esta tesis, consiste en que el robot pueda actualizar y modificar la información del sistema de señalización usando información que las personas en el entorno puedan proporcionarle. Para esto usará la comunicación verbal para el proceso de interacción humano-robot, ya que con este tipo de comunicación, es posible intercambiar información más específica y enriquecedora a la hora de escribir y/o complementar la información de las señales de navegación en el entorno.

### Tipo de información que el robot requiere

El tipo de información que el robot debe adquirir dependerá del contenido de la señal del entorno que el robot va a escribir o modificar. En la sección 3.3.1, se ha comentado el tipo de información que contienen las señales del entorno para el modelo de navegación presentado en esta tesis. Como se ha dicho, la información básica de una señal debe responder al menos a una de las siguientes cuestiones: ¿dónde estoy?, ¿cómo voy? y ¿hacia dónde puedo ir?. Para ello el robot necesitaría saber el contenido de los elementos básicos de una señal.

En 4.2, se ha explicado el diseño y contenido de una señal en el sistema de señalización. Si el robot pregunta y recibe las indicaciones de una persona para señalar el entorno, debe consultar y responder a ésta con un lenguaje natural y no mediante un lenguaje técnico, es decir, no puede hacer preguntas del tipo "*Dime el contenido de la etiqueta < place > del fichero XML.*" Es por eso, que los diálogos entre el humano y el robot, deben ser comprensibles por ambas partes, por un lado que la persona pueda responder con un lenguaje natural y coloquial y que pueda entender lo que el robot esté consultando, y por el otro que el robot pueda comprender la información recibida, para posteriormente procesarlos y escribir la señal.

A continuación se presentan los dos posibles casos que se pueden presentar al iniciarse el proceso de interacción.

**Caso 1** Un caso sencillo de adquisición de datos consistiría en un diálogo simple, que comenzaría cuando el robot encuentra una señal vacía. Si el robot tiene próxima a una persona para solicitarle información, el robot puede tomar la iniciativa y consultar mediante interacción, los datos referentes al lugar en el que se encuentra y escribirlo en la señal. Dependiendo de la información que el robot requiera, habrán dos posibles tipos de diálogos:

#### ■ Diálogo 1. ¿Dónde estoy?

En este diálogo el robot preguntará por su ubicación, consultando los elementos básicos de un *lugar*: el nombre y su identificador.

Ejemplo:

Robot: ¿Cómo se llama este lugar?
Usuario: Este es el Laboratorio de Robótica
Robot: Ok, dame mas información sobre este sitio. ¿Cuál es el identificador de este lugar?
Usuario: Es el 1.3.C.14
Robot: ¿Tienes mas información de este lugar?
Usuario: No, no la tengo.
Robot: Muy bien, gracias por tu ayuda.

### ■ Diálogo 2. ¿Hacia donde puedo ir?

En este diálogo, el robot pregunta al usuario por las conexiones del lugar. El usuario podrá darle el número de conexiones que conozca. Para el ejemplo, el usuario indica sólo una conexión. El robot durante el diálogo solicita datos de los dos elementos claves de la conexión: el *lugar* y la *acción*.

Ejemplo:

Robot: ¿Desde aquí puedo ir a otros sitios?
Usuario: Si Maggie.
Robot: Muy bien, estoy atenta a tus indicaciones Dime, ¿hacia donde puedo ir?
Usuario: Puedes ir al Laboratorio de Robótica 2
Robot: ¿Cuál es el identificador de ese lugar?
Usuario: Es el 1.3.C.13
Robot: ¿Qué debo hacer?
Usuario: Para ir ahí, debes seguir por la derecha.
Robot: Hay otro sitio al que pueda ir desde aquí?
Usuario: No, no conozco ningún otro sitio.
Robot: Muy bien, gracias por tu ayuda. Voy a almacenar la información.

También es posible que el robot, durante la conversación, combine los diálogos anteriores, es decir, que el robot solicite información del lugar donde se encuentre y de las conexiones o lugares donde puede ir desde ese sitio con las acciones correspondientes. El robot preguntará hasta que el usuario le diga que no posee más información, incluso, es posible que el usuario le proporcione información incompleta, por ejemplo, que sepa el nombre del lugar pero no su identificador, o que el usuario no sepa el nombre del lugar actual pero si hacia donde puede ir el robot.

**Caso 2** Otro posible caso consiste, en que sea una persona la que tome la iniciativa e indique al robot el contenido de las señales. La persona puede indicarle al robot donde se encuentran las señales, una vez que el robot esté en la señal (en este una etiqueta RFID), el usuario procede a proporcionarle al robot la información del lugar, a través de un lenguaje natural, que el robot procesará y modelará para almacenarla en la señal.

**■ Diálogo 3. El usuario toma la iniciativa.**

Usuario: Maggie, voy a darte las indicaciones de éste lugar.  
Éste lugar es el laboratorio de Robótica 3.

Robot: Entendido.

Usuario: El identificador de este lugar es el 1.3.C.12

Robot: Muy bien, ¿tienes mas información de este lugar?

Usuario: No, no la tengo.

Robot: Entonces guardaré la información.

En éste tipo de interacción, es posible que además de indicaciones verbales referentes a una señal, hayan órdenes para ejecución de acciones de movimiento.

**■ Diálogo 4. El usuario indica al robot acciones de movimiento.**

Usuario: Maggie, sígueme.

Robot: Muy bien, te sigo.

Robot: He detectado una etiqueta.

Usuario: Maggie, voy a darte las indicaciones de éste lugar.  
Éste lugar es el laboratorio de Robótica 3.

Robot: Entendido.

Usuario: El identificador de este lugar es el 1.3.C.12

Robot: Muy bien, ¿tienes mas información de este lugar?

Usuario: No, no la tengo.

Robot: Entonces guardaré la información.

Usuario: Ahora sígueme, seguiré mostrándote el lugar.

Usuario: Maggie, voy a darte las indicaciones de éste lugar.  
Éste lugar es el laboratorio de Robótica 2.

Robot: Entendido.

Usuario: El identificador de este lugar es el 1.3.C.13

Robot: Muy bien, ¿tienes mas información de este lugar?

Usuario: No, no la tengo.

Robot: Entonces guardaré la información.

Como hemos visto en el diálogo anterior, el usuario puede guiar al robot hacia las señales (etiquetas RFID) a través de órdenes de movimiento. El robot ejecutará la habilidad indicada, a través de frases como "Sígueme" o "Sigue por la derecha", hasta que encuentre una señal y el usuario le proporcione la información del lugar.

### 5.2.2. Adquisición de la información por deducción a través de señales previas

Una manera de adquirir la información de una señal para realizar el proceso de señalización es usar otras señales colocadas en el entorno como fuente de información. Si el robot se encuentra navegando por el entorno y encuentra señales "incompletas", de manera autónoma, podrá ir completando la información de las etiquetas de acuerdo a la información previa (inmediata) que haya adquirido.

Se dice que la información de una señal es incompleta, cuando falta uno o más elementos que conforman una señal:

- El lugar (*Place*).
- Una conexión o conjunto de conexiones.

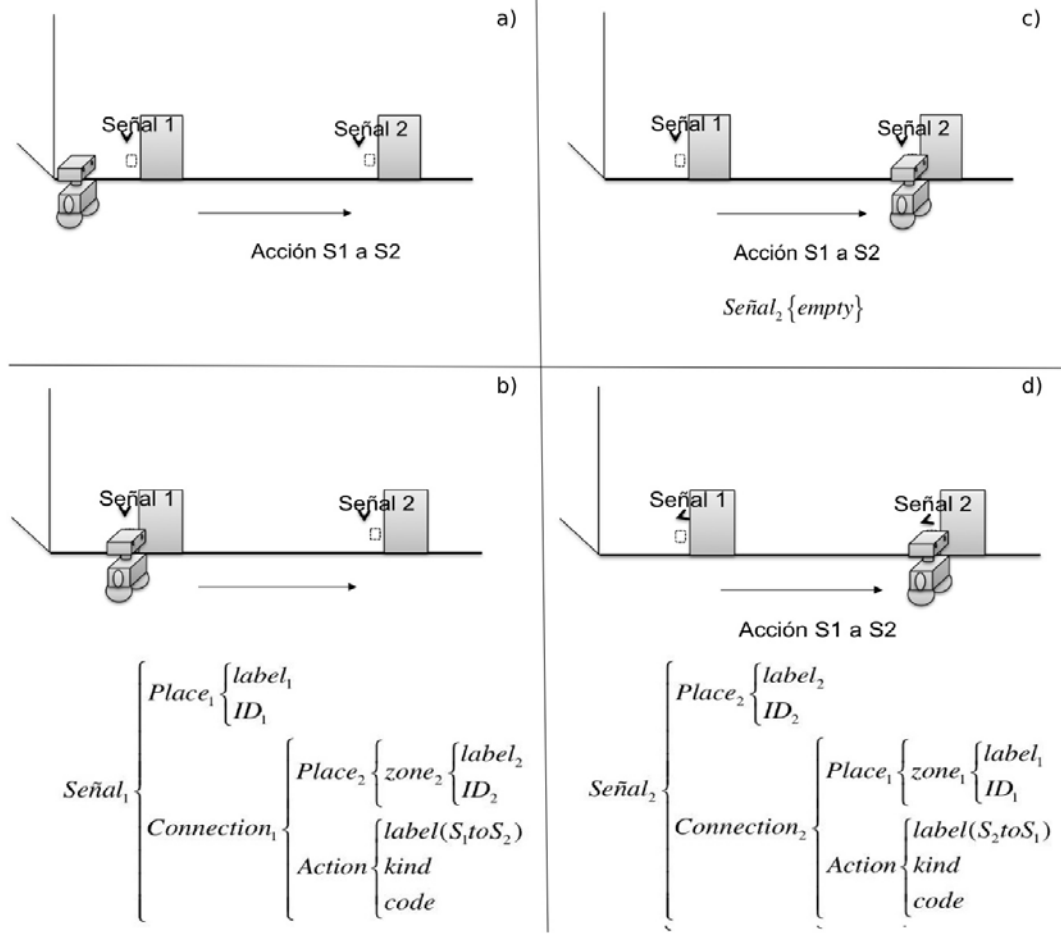
También decimos que es incompleta, cuando falta uno o más componentes en un elemento de la señal. Por ejemplo, en un *lugar* específico, el elemento *label* o el *ID*.

Como se ha dicho anteriormente, la información de la señal podrá completarse con la información previa inmediata que haya adquirido el robot, por lo que para éste tipo de adquisición de información es necesario usar la Memoria a Corto Plazo del sistema.

El procedimiento es el siguiente:

- Inicialmente el robot lee información de una señal  $S_1$ .
- El robot seguirá explorando.
- Al encontrar la siguiente señal  $S_2$ , el robot lee la información. Si es incompleta, el robot la complementa con la información que ha leído de la señal previa. El robot sabe cuál es la etiqueta de donde proviene por lo que en la próxima etiqueta puede asignar el valor de el elemento *lugar* de la *conexión*. El elemento *acción* se puede completar si la acción realizada por el robot para llegar a esa señal ha sido simple, por ejemplo, si el robot realizó la acción previa  $S_1 \text{ to } S_2$  "seguir contorno por la izquierda", es posible deducir la acción  $S_2 \text{ to } S_1$  como la acción opuesta "seguir contorno por la derecha".

En la figura 5.1, se explica el proceso de adquisición de información mediante deducción a través de una señal previa. En el recuadro 5.1a se muestra a un robot recorriendo un pasillo. Al encontrar la señal  $S_1$  (5.1b), éste lee la información y ejecuta la acción  $S_1 \text{ to } S_2$  cómo le ha indicado la conexión  $Connection_1$ . Posteriormente el robot encuentra la señal  $S_2$  (5.1c), al leerla está vacía, por lo que el robot en base a



**Fig. 5.1:** Ejemplo del proceso de adquisición de información mediante deducción a través de una señal previa: Robot navegando por un pasillo a través de dos señales.

la información de la señal previa  $S_1$  procede a completarla. El lugar está dado por el nombre de la conexión  $Connection_1$  leído previamente. El elemento conexión lo complementa colocando la información de la señal  $S_1$  (5.1d).

### 5.3. Diseño del sistema de diálogo para la adquisición y escritura de de datos de una señal

En las secciones anteriores, se ha explicado el tipo de información y la manera en que el robot obtendrá la información de las señales. Particularmente en ésta tesis, el proceso de adquisición de información se ha enfocado en la interacción humano-robot mediante diálogos. Para lograr este propósito, en la implementación del proceso de señalización que se presenta en éste trabajo, se han utilizado dos habilidades desarrolladas e integradas en la arquitectura software del robot [de Gorostiza, 2010, Gorostiza and Salichs, 2011, Alonso-Martin and Salichs, 2011]:

- **Habilidad de reconocimiento automático del habla *asrSkill*:** Ésta habilidad es la encargada de tomar los datos del habla del usuario y procesarlos para su uso en el sistema. Cuenta con un motor de reconocimiento automático de habla (Loquendo<sup>2</sup>, que permite un reconocimiento continuo del habla, alto grado de acierto y confianza en el reconocimiento, rapidez de respuesta y además permite el uso de gramáticas dinámicas (en tiempo de ejecución).
- **Habilidad de síntesis de texto a voz *ettsSkill*:** Ésta habilidad recibe como entrada un texto (generado por alguna habilidad en la arquitectura) y lo sintetiza, considerando parámetros como velocidad, entonación, volumen, frecuencia y algunas otras características (como parámetros emocionales) que permiten mayor expresividad al robot.

Para los diálogos dinámicos, el sistema usa un gestor de diálogo que está basado en un intérprete *VoiceXML* [de Gorostiza, 2010].

*VoiceXML* (VXML) es un estándar XML de W3C<sup>3</sup> que especifica el formato para diálogos de interacción humano-máquina. Es un lenguaje de etiquetado que permite crear diálogos con los que se puede interactuar a través de entradas de voz. Éste intérprete se encarga de convertir habla en texto y para ello utiliza diferentes gramáticas para el reconocimiento del habla.

### 5.3.1. Definición de las gramáticas para la adquisición de información en la señalización

Según la RAE, el término *Gramática* se define como la “Ciencia que estudia los elementos de una lengua y sus combinaciones”. En el contexto de reconocimiento automático de habla, en las gramáticas se establece un conjunto de elementos que conforman el lenguaje que el reconocedor de habla recibirá para poder relacionar las palabras con contenidos semánticos [W3C, 2004].

La gramática, en este trabajo, corresponde a las frases y/o palabras concretas que el robot reconocerá para asociarlo con la información del entorno y las tareas de navegación. La gramática dependerá del nombre del sitio donde se puede encontrar el robot, los sitios a los que puede ir y las acciones que debe realizar.

#### Frases posibles

El diseño de la gramática está basado en las posibles frases que el usuario podría proporcionar al robot para indicarle la información de una señal. Las frases deben responder a las preguntas básicas: ¿Dónde estoy?, ¿Hacia donde puedo ir? y ¿Cómo puedo ir?. Para los diálogos de señalización se tienen diferentes tipos de frases posibles:

- **Frases que definen un lugar concreto:** Se refiere al nombre específico de los lugares. Dependerá del sitio donde navegue el robot. Por ejemplo: “*Despacho del Director*”, “*Laboratorio de Robótica*”, “*Aula informática*”, etc.

<sup>2</sup> Loquendo: Sociedad global. Soluciones automáticas y tecnología vocal <http://www.loquendo.com/es>

<sup>3</sup> The World Wide Web Consortium (W3C) <http://www.w3.org/>



- **Frases para definir la ubicación actual:** Del tipo “*Estas en*”, “*Esto es*”, “*Aquí es*”.
- **Información única de un lugar:** Esto se refiere a los identificadores únicos de cada lugar. Por ejemplo: Para el *Laboratorio de Robótica 2* el ID será “*1 punto 3 C 13*”.
- **Frases direccionales o relacionados con las conexiones:** Del tipo: “*Puedes ir a*”, “*hacia*”, “*por ahí queda*”, “*hacia allá*”, etc.
- **Frases que definen una acción concreta:** Están relacionadas con las habilidades de navegación que el robot debe realizar para ir de un sitio a otro. Por ejemplo: “*Avanza por la derecha*”, “*Seguir por la izquierda*”, etc.

### Definición del lenguaje

Basándonos en las frases posibles que el usuario dirá al robot, se procede a definir el lenguaje con las diferentes reglas gramaticales que el reconocedor automático de habla del robot relacionará con la tarea de señalización. Éstas son:

**\$lugares** Corresponde a los nombres de los lugares o zonas que se van a señalar y/o hacia donde el robot puede ir.

**\$ids** Son los diferentes identificadores alfanuméricos de los lugares o zonas del entorno.

**\$acciones** Las acciones son los movimientos o tareas que debe realizar el robot para llegar a una zona o lugar.

**\$adjetivos** Son los diferentes adjetivos direccionales que complementan la información de la ubicación y de las conexiones.

**\$tipodeaccion** Con ésta regla se determinará si la acción es geométrica o no.

**\$op** Proporciona al robot respuestas genéricas de afirmación o negación.

### Definición Semántica

La gramática semántica permite al reconocedor devolver valores relacionados con el significado de las frases que reconoce. Según el estándar ABNF, es posible incorporar a la gramática una interpretación semántica dentro de cada regla gramatical:

$$< @attribute = value >$$

Donde *@attribute* es la variable que almacena el valor semántico dado por *value*.

Por ejemplo, en la siguiente regla gramatical:

```
public $op= (“si”:si|“no”:no|“okey”:si|“negativo”:no {@option $value}
```

La variable `@option` almacenará el valor semántico de la regla gramatical `$op` (“*si*”, “*no*”, “*negativo*”...). Ésta variable será la que usará el Gestor de Diálogo para almacenar la frase reconocida y enviarla posteriormente a la Habilidad de Señalización (ver sección 6.8).

La siguiente gramática corresponde a la descripción del lenguaje para la comunicación de información sobre lugares y conexiones durante la tarea de señalización. La construcción de las reglas gramaticales sigue el estándar ABNF (Augmented Backus-Naur Form) [Crocker and Overell, 2008]. Para éste ejemplo, la señalización corresponde a la Tercera Planta del Edificio Betancourt de la Universidad Carlos III de Madrid.

```
#ABNF 1.0 ISO-8859-1;

language es-ES;
tag-format <loq-semantics/1.0>;

public $root = $waymarking;

public $waymarking = $lugares | $adjetivos | $ids | $op | $acciones |
                    $codigos | $tipodeaccion;

public $lugares = ("edificio betancourt" | "tercera planta" |
"departamento de ingeniería de sistemas y automática zona c" |
"laboratorio de maggie" | "laboratorio de robotica 2" |
"departamento de ingeniería de sistemas y automática zona b" |
"departamento de ingeniería eléctrica" |
"laboratorio de robótica 3" | "servicios" | "televisión" | "tele" |
"tercera planta" | "zona" ){<@lugares $value>};

public $ids = ("uno" | "1 punto 3" | "1 punto 3 c" | "1 punto 3 c 12"
| "1 punto 3 c 13" | "1 punto 3 b" | "1 punto 3 d" | "1 punto 3 c 14"
| "1 punto 3 c s" | "1 punto 3 c 12 tv" | ){<@ids $value>};

public $op = ("si":si | "no":no | "okey":si | "vale":si | "negativo":no
| "de acuerdo":si | "incorrecto":no ){<@op $value>};

public $acciones = ("sigue el contorno por la derecha"
| "sigue el contorno por la izquierda"
| "usa la habilidad ir a un punto"){<@acciones $value>};

public $adjetivos = [("esto es"|"estas en"|"aquí es"|"puedes ir a"|"hacia")|$lugares;

public $codigos = ("0" | "1" | "2" | "3"){<@codigos $value>};

public $tipodeaccion = ("si" | "no"){<@tipodeaccion $value>};
```

## 5.4. Discusión

La facilidad que proporciona la tecnología RFID de crear y modificar la información que compone una señal es uno de los factores que facilitan la incorporación de nuevas señales en el entorno, independientemente del conocimiento que una persona tenga sobre ésta tecnología. A través de la interacción humano-robot, un robot social podrá crear y modificar las señales del entorno siguiendo las indicaciones que le proporcione un humano a través de un lenguaje coloquial. En este capítulo se ha propuesto un sistema de escritura de las señales del entorno, de modo que el robot pueda interactuar con las personas para adquirir la información de las señales que se quieren escribir y así automatizar el proceso de señalización. Para ésto fue necesario diseñar el sistema de diálogo donde tanto el humano como el robot puedan tomar la iniciativa en la interacción.



## 6. SISTEMA EXPERIMENTAL

### 6.1. Introducción

El presente capítulo expondrá el diseño y desarrollo del sistema experimental en el cual se ha puesto en marcha el sistema de navegación que se presenta en esta tesis. Inicialmente se hablará de la plataforma robótica donde se ha implementado este trabajo, de los sistemas de Identificación por Radio Frecuencia y su incorporación en el robot, tanto en términos de hardware como en software. Posteriormente se hablará de la estructura de la información de una señal para la navegación y su almacenamiento en etiquetas RFID así como la colocación de éstas como señales en el entorno. Seguidamente se explicará la implementación del Algoritmo de Navegación y su integración en el robot. Finalmente se explicará el desarrollo e implementación de la Habilidad de Señalización.

### 6.2. El Robot Maggie

La interacción humano-robot juega un papel muy importante en la incorporación de los robots en la vida diaria. Los robots hasta ahora han sido utilizados en entornos industriales, estructurados y con bajo nivel de autonomía, la interacción humano-robot nos lleva a tareas más cercanas a usuarios sin conocimientos técnicos, siendo su aplicación en actividades como asistencia en el hogar, asistencia a personas discapacitadas, robots que sirvan de compañía, asistencia personal, entre otros. Maggie es el robot social desarrollado en el RoboticsLab de la Universidad Carlos III de Madrid, para investigación en la interacción humano-robot [Salichs et al., 2006].

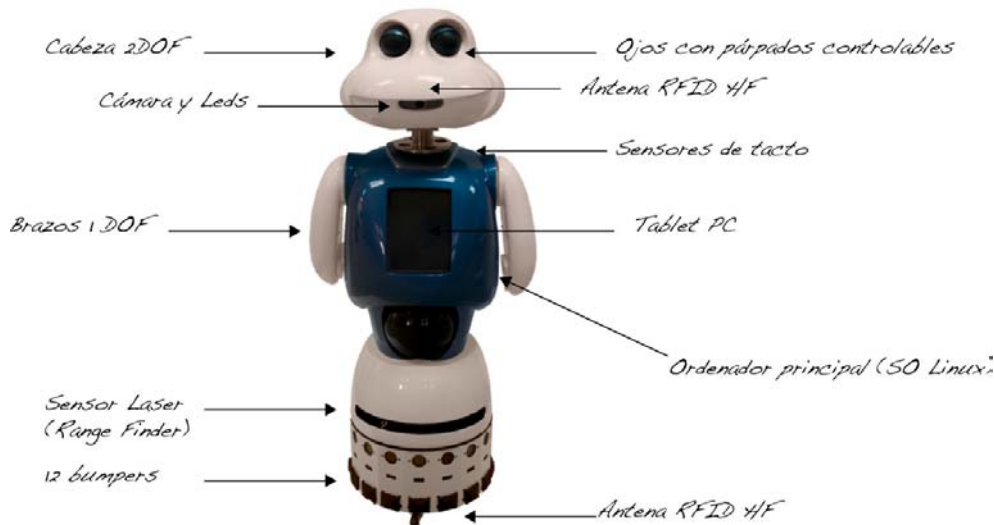
#### 6.2.1. Características del robot

Como se muestra en la figura 6.1, Maggie tiene un diseño artístico con las siguientes características:

**Atractivo:** La apariencia física es importante al interactuar con el robot. Maggie posee una apariencia amigable, con 1.35 m de altura y colores agradables a la vista.

**Expresividad:** Maggie no posee expresiones faciales pero puede mover los párpados, tiene una serie de leds azules en la boca que están sincronizados con la voz.

**Multimodalidad:** Posee diferentes mecanismos de interacción, tiene varios sensores de tacto distribuidos en diferentes partes del robot, una cámara para procesamiento



**Fig. 6.1:** El Robot Social Maggie.

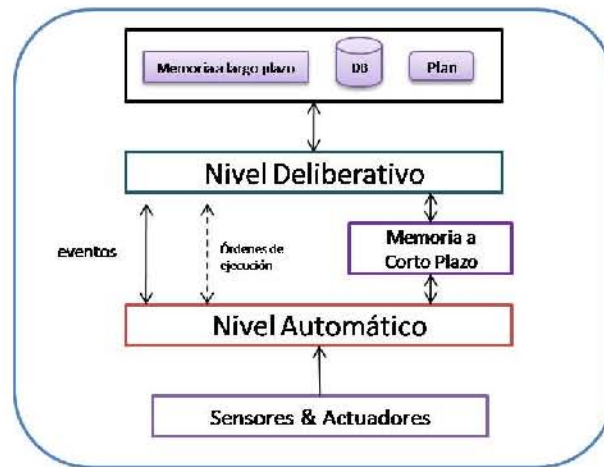
de imágenes, para interacción por voz posee altavoces y mediante un micrófono obtiene la información vocal del usuario.

**Movilidad:** El movimiento del robot se consigue a través de una base con ruedas y guiado diferencial.

El robot está equipado con 12 sensores de contacto y un láser Sick LMS 200. La cabeza del robot posee una forma antropomórfica con 2 grados de libertad. Además tiene 2 brazos de 1 grado de libertad y varios sensores capacitivos distribuidos por el cuerpo que actúan como sensores de tacto. Dentro de Maggie hay 2 ordenadores interconectados entre sí a través de una red Ethernet y con ordenadores externos a través de WiFi 802.11, uno de los ordenadores es un tablet PC para tareas de interacción y el otro un ordenador con el sistema operativo Linux que maneja el software de los diferentes elementos físicos del robot, y procesa el control y habilidades del robot.

### 6.2.2. Arquitectura Software del Robot

La arquitectura software del robot Maggie está basada en la arquitectura Automático-Deliberativa AD [Barber, 2000, Rivas et al., 2007, Rivas, 2010], desarrollada en el RoboticsLab de la UC3M. Esta arquitectura está inspirada en los procesos mentales de los humanos. El nivel deliberativo está asociado con procesos reflexivos, es decir, los procesos que requieren de capacidad de razonamiento. El nivel automático se refiere a los procesos automáticos del sistema, esta compuesto por los módulos que interactúan con los sensores y actuadores. La comunicación entre el nivel Automático y el nivel Deliberativo es bidireccional. Ambos niveles se comunican a través de la Memoria a Corto Plazo (MCP). La MCP guarda la información más importante suministrada por los sensores o por habilidades. Ambos niveles están compuestos por *habilidades* (Fig. 6.2).



**Fig. 6.2:** Arquitectura Automático-Deliberativa AD.

### Definición de Habilidad

Una habilidad es el elemento básico de la arquitectura AD. Consiste en la capacidad del robot de realizar un razonamiento, de procesar información o de llevar a cabo una acción. Las habilidades pueden ser activadas por otras habilidades o por un secuenciador de órdenes, devolviendo datos o eventos al elemento que la haya activado o a habilidades que estén interesadas en la información suministrada.

La comunicación entre habilidades se realiza mediante un sistema Manejador de Eventos y el uso de la Memoria a Corto Plazo (MCP).

- Cada habilidad puede notificar su estado a través de un evento, enviar mensajes, o notificar que ha escrito datos en la MCP.
- La activación y bloqueo de una habilidad puede ser realizada por otra habilidad o por un secuenciador de habilidades que envíe un evento para este propósito.
- Una habilidad que emite eventos, no conoce de antemano si existen o no otras habilidades que estén dispuestas a gestionar el evento emitido, ya que el Manejador de Eventos sigue el paradigma *Publisher/Suscriber* [Gamma et al., 1994].
- Una habilidad puede suscribirse a un evento y, al recibirlo, gestionarlo ejecutando un proceso en función de la información recibida en el evento.
- Las habilidades comparten información a través de la Memoria a Corto Plazo, una habilidad puede escribir datos y notificar mediante un evento a otras habilidades para que lean y usen la información.

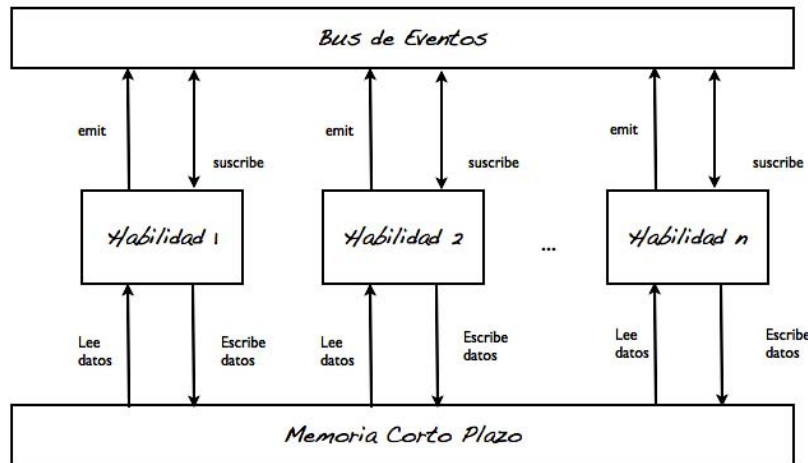
En términos de ingeniería del software, una habilidad es un objeto que encapsula los datos y procesos que describen el comportamiento global de la tarea o acción que debe realizar el robot. El diseño basado en objetos facilita el desarrollo del software debido a que permite la reutilización de componentes, la herencia y el polimorfismo. La clase habilidad tiene un método denominado *proceso* que define el lazo de control

de la habilidad. Este método es redefinido en cada subclase habilidad. También incluye un método que la activa o desactiva desde un objeto o aplicación externa.

Basándonos en las premisas anteriores, la *Habilidad de Navegación Mediante Señales*, se desarrollará en base a 3 condiciones básicas:

1. Heredar de la clase base *Habilidad* de la Arquitectura AD.
2. Compartir y recibir información de otras habilidades del sistema (uso de la *Memoria a Corto Plazo*).
3. Advertir y recibir eventos y notificaciones de otras habilidades (uso del *Manejador de Eventos*).

La figura 6.3 muestra los elementos que intervienen en la comunicación entre habilidades en la Arquitectura AD.



**Fig. 6.3:** Esquema de comunicación entre habilidades en la Arquitectura AD.

## 6.3. Sistema RFID implementado en el robot

### 6.3.1. Hardware RFID en el Robot Maggie

Para el desarrollo e implementación de las habilidades fue necesario seleccionar un lector y antena RFID que cumplieran el objetivo final del sistema de navegación y que a su vez, pudiera ser fácilmente incorporado en el robot Maggie. Dos aplicaciones usan RFID en el robot personal Maggie: una para reconocimiento de objetos y otra para soporte en tareas de localización y navegación [Corrales, 2008, Corrales and Salichs, 2009].



La elección se ha hecho tomando en cuenta la frecuencia de operación, los tipos de etiquetas, tamaño del lector y las características de software y hardware (ver conceptos básicos de los sistemas RFID en el apéndice B).

1. **Frecuencia de operación:** Es una de las variables más importantes al realizar el diseño de un sistema RFID. Para seleccionar el tipo de frecuencia se debe tomar en cuenta la distancia de lectura, esto dependerá del tipo de aplicación así como de las condiciones de operación. En el cuadro 6.1 se muestra los diferentes rangos de frecuencia con los rangos de lectura y el tipo de etiquetas RFID correspondiente.

Rango de Frecuencia	Distancia de Lectura	Tipos de Etiquetas
Baja Frecuencia (125-134 kHz)	Hasta 10 cm	Pasivas
Alta Frecuencia (13.56 MHz)	Entre 10 y 100 cm	Pasivas o activas
UHF (860-960 MHz)	Entre 60 cm y 30 m	Pasivas o activas

**Tab. 6.1:** Rango de Frecuencias RFID.

2. **Tipos de Etiquetas:** Las etiquetas las podemos seleccionar dependiendo de varios factores, por ejemplo: tipo de alimentación, tipo de memoria, tamaño y distancia de lectura. Los cuadros 6.2 y 6.3 muestran las características principales de las etiquetas según su alimentación y memoria.

Tipo de Etiqueta	Alimentación	Distancia de Lectura
Activas	Batería propia	30 m aprox.
Semi-Activas	Batería propia	10 y 15 m
Pasivas	Sin alimentación	3 cm y 9 m aprox.

**Tab. 6.2:** Etiquetas RFID según su alimentación.

Tipo de Memoria	Características
Read Only	De solo lectura, el identificador está grabado de fábrica.
WORM (Write Once Read Many)	El usuario escribe la información una vez y puede leerla varias veces.
Lectura/escritura programable	Para aplicaciones que requieren reutilización de etiquetas.

**Tab. 6.3:** Etiquetas según el tipo de memoria.

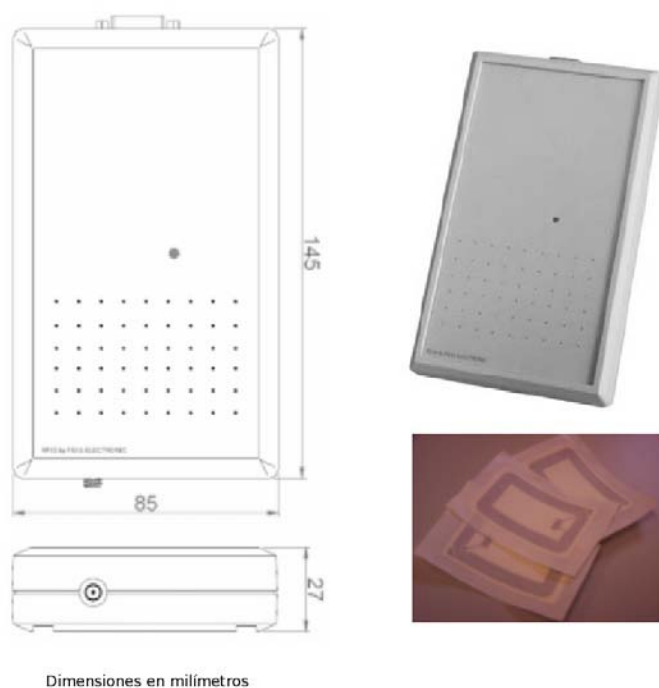
### Primera selección

En un principio se seleccionó un lector con una frecuencia de 13,56 MHz, considerando que una distancia de lectura de etiquetas entre 15 y 20 cm es suficiente para

que el robot se aproxime y pueda leer la información de las etiquetas. No se seleccionó un lector de baja frecuencia ya que el rango de lectura es muy corto (de contacto). En el momento de la selección, los lectores UHF existentes en el mercado eran de un tamaño considerable, lo que hacía difícil la incorporación en el robot. En el caso de localización y navegación, se ha optado por colocar el lector en la base del robot y ubicar las etiquetas RFID en el suelo, el robot las leerá cuando pase sobre éstas.

Por otra parte una antena externa es complicada de incorporar al robot, así que se debe seleccionar un lector con antena integrada. Las etiquetas se han seleccionado pasivas, ya que no necesitan fuente de alimentación, además de ser las de más bajo coste.

De los diferentes lectores buscados en el mercado se ha seleccionado el lector D ISC.MR101-USB 13.56 MHz de la empresa FEIG Electronic <sup>1</sup>. Es un lector de corta distancia (18 cm aprox.), permite leer etiquetas pasivas de diferentes estándares (ISO15693, I-Code y Tag-it HF e I-Code EPC), sus dimensiones (145x85x31mm) son adecuadas para el robot (Figura 6.4). La interfaz es USB, además que posee los drivers y bibliotecas de software para el sistema operativo Linux, lo que implica mayor facilidad de integración en la arquitectura de software del robot.

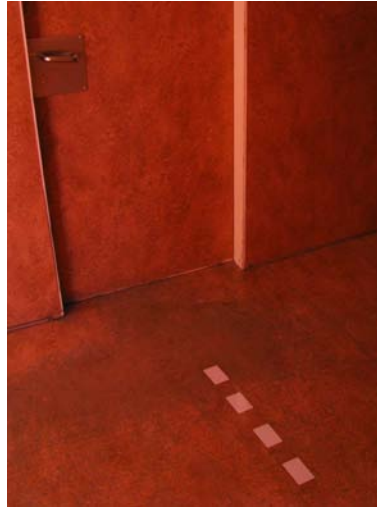


**Fig. 6.4:** Lector RFID de la empresa FEIG Electronic y etiquetas pasivas.

**Inconvenientes y soluciones** El principal inconveniente con este hardware es la distancia de lectura de las etiquetas (18 cm máx.). Por seguridad en la navegación

<sup>1</sup> FEIG Electronic <http://www.feig.de/>

para el robot, las etiquetas se han colocado en el suelo, de manera que el robot lea las etiquetas cuando pase por encima de ellas, el problema es que la antena del lector D ISC.MR101-USB es de polarización lineal por lo que sólo podría leer la etiqueta si pasa exactamente sobre ella  $\pm 10$  cm de error de radio. Es por ese motivo que se ha optado por colocar 4 etiquetas dispuestas linealmente y equidistantes entre sí (Fig. 6.5). Las etiquetas tendrán almacenada la misma información, de manera que el robot pueda detectar una u otra sin necesidad de tener exactitud de movimiento durante la navegación.



**Fig. 6.5:** Etiquetas HF en el entorno.

### **¿Y si el robot requiere escribir etiquetas RFID para almacenar información?**

Para leer etiquetas RFID la selección del hardware es adecuada, como se reflejará más adelante en los resultados experimentales de la lectura de señales para el sistema de navegación. Sin embargo, en el sistema de navegación, cabe la posibilidad que el robot pueda escribir y/o modificar la información de las etiquetas, bien sea como una manera de automatizar el proceso de señalización o para corregir posibles errores en las señales (Capítulo 5).

Para escribir las etiquetas alineadas, el robot tendría dificultades, ya que, si el robot detecta una de las cuatro etiquetas de la línea, sólo podría escribir en la que ha detectado, quedando las otras tres con la misma información sin modificaciones.

Es por eso que se ha decidido usar lectores RFID UHF, porque proporcionan mayor rango de lectura y es posible continuar con la tecnología de etiquetas pasivas [Yébenes et al., 2011]. Aunque actualmente en el mercado, el tamaño de los lectores sigue siendo un poco grande, se ha optado por el lector ID ISC.MRU200i-USB (865-928 MHz) con dimensiones 200 mm x 110 mm x 60 mm (Fig. 6.6). Usa etiquetas pasivas y el rango de lectura aproximado es de 80 cm.

Se han colocado en el robot dos lectores RFID UHF, uno a cada lado del robot (Fig. 6.7), así es posible colocar las etiquetas RFID en las paredes del entorno y el robot



**Fig. 6.6:** Lector UHF RFID de la empresa FEIG Electronic.

podrá navegar de una manera más segura. Con el rango de lectura que se tiene y la polarización de la antena más amplia, es posible entonces colocar una sola etiqueta (en la pared, por ejemplo) como señal y no cuatro alineadas en el suelo, como se implementó inicialmente, así el problema de la escritura de etiquetas es solucionado.



**Fig. 6.7:** El robot Maggie con las antenas UHF.

### 6.3.2. Habilidades RFID

Para desarrollar las habilidades de identificación mediante RFID, se han incorporado al robot dos tipos de lectores RFID con antena incorporada (HF y UHF), de manera tal que el robot pueda leer y escribir información en etiquetas RFID que se encuentren próximas en el entorno. Los lectores RFID tienen diferentes propósitos:

detección e identificación de objetos y servir como soporte en las tareas básicas de localización y navegación en entornos interiores.

Se han realizado dos habilidades generales para identificación con RFID para lectura y escritura de datos, independientemente de su posterior aplicación (reconocimiento de objetos, medicinas, navegación, etc.) [Corrales, 2008]. El sistema RFID del robot debe acceder a las funciones básicas del lector y entonces definir las funciones de detección, lectura y escritura de las etiquetas RFID, usando los diferentes componentes de la arquitectura AD: sistema manejador de eventos, la Memoria a Corto Plazo y la clase básica habilidad.

### Casos de uso

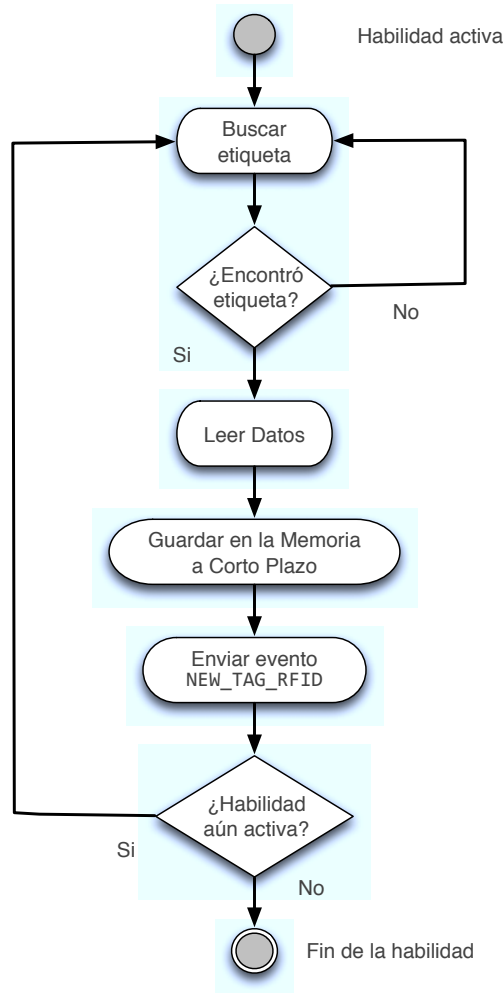
El lector RFID se mantiene en ejecución constante detectando las etiquetas que se encuentran en su entorno inmediato. Las habilidades RFID notificarán a otras habilidades del sistema sobre la detección, lectura y escritura de etiquetas. Estas habilidades cliente se suscriben al evento `NEW_TAG_RFID`, que indica que el lector RFID ha encontrado una etiqueta y almacenado los datos en la MCP. También es posible que otras habilidades envíen datos a las habilidades RFID, para que sean escritos en las etiquetas.

En la figura 6.8 se muestra el diagrama de flujo de la habilidad de lectura RFID. Cuando la habilidad es activada por un secuenciador u otra habilidad, comienza a buscar etiquetas en el entorno. Si la etiqueta es detectada realiza la lectura de los datos almacenados en la etiqueta y los guarda en la Memoria a Corto Plazo. Inmediatamente envía el evento `NEW_TAG_RFID`, para que las habilidades interesadas o suscritas a éste, puedan realizar las acciones correspondientes. Si la habilidad aún se encuentra activa vuelve a buscar etiquetas y hacer de nuevo el proceso anterior, si no está activa finaliza su ejecución o se bloquea esperando órdenes.

La habilidad de escritura es más simple. La operación de escritura es puntual. Cuando recibe la orden de escritura la habilidad activa la función *buscar etiqueta*. Si encuentra la etiqueta escribe los datos y finaliza la función escritura y si no la encuentra la sigue buscando hasta que la encuentre, hasta que pase un tiempo máximo de búsqueda o hasta que reciba la orden de interrupción de la operación (Fig. 6.9).

Los lectores RFID usados en esta tesis poseen una biblioteca de funciones realizadas por la empresa FEIG Electronic, están desarrolladas en C++ para facilitar el desarrollo de programas para los lectores OBID-RFID. La biblioteca ID FEDM soporta todos los lectores de la familia OBID y será una capa adicional en el desarrollo de las habilidades RFID.

En la figura 6.10 se muestra un esquema básico de la arquitectura de capas inicial del sistema RFID. Para desarrollar la habilidad se realizó una clase intermedia o API (Application Programming Interface), para acceder a los procedimientos básicos de búsqueda, lectura y escritura de etiquetas de la biblioteca OBID.

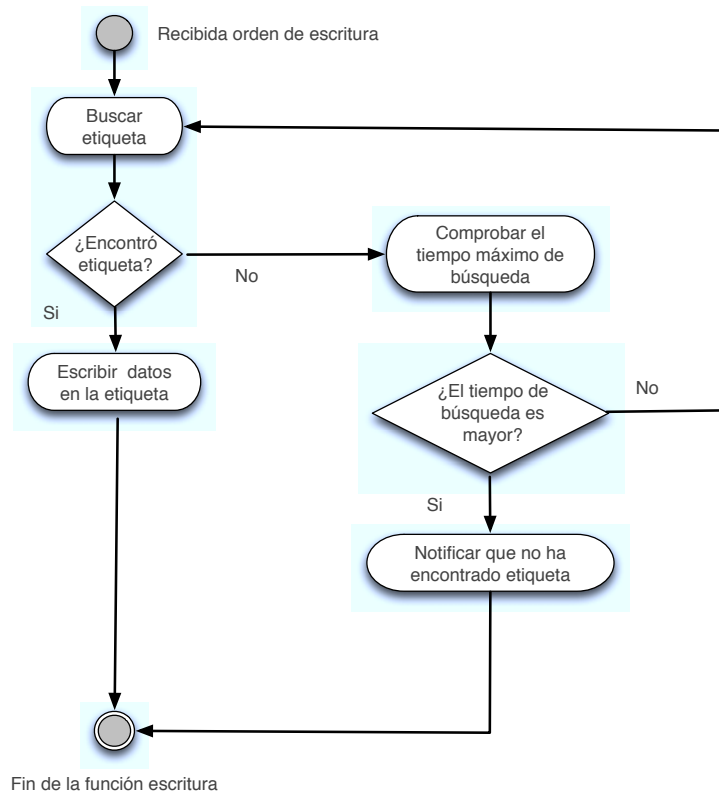


*Fig. 6.8: Diagrama de flujo de datos de la habilidad de lectura RFID.*

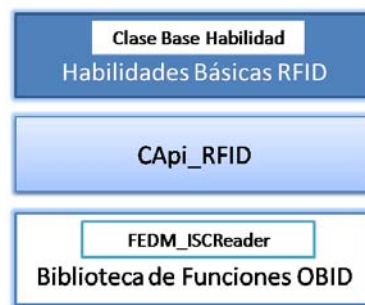
### 6.3.3. Habilidad de Lectura de Datos CRFID\_ReadSkill1

Esta habilidad es la encargada de obtener los datos de las etiquetas RFID usando las funciones de la interfaz de programación API RFID. Es una clase que hereda de la clase base CHabilidad, uno de los componentes principales de la arquitectura AD, debe compartir la información mediante la Memoria Corto Plazo y notificar a las demás habilidades del sistema mediante el Sistema Manejador de Eventos. Basado en estas precondiciones de diseño se han seleccionado los siguientes atributos de la clase:

- **CEventManager emisor:** Es el objeto emisor de eventos, usado para notificar a las habilidades que ha sido detectada y leída una nueva etiqueta RFID y que los datos han sido guardados en la MCP.
- **Tdatos\_etiqueta:** Es el formato en que se guarda la información de las etiquetas en la memoria compartida. Es una estructura con un identificador de dato que indica que es un dato de lectura.



**Fig. 6.9:** Diagrama de flujo de datos de la habilidad de escritura de etiquetas RFID.



**Fig. 6.10:** Arquitectura software básica del sistema RFID.

- **Capi\_RFID lector:** Objeto para manejar las funciones básicas de lectura.
- **ID\_Lector:** Número identificador único del lector RFID. Con esto se consigue activar la habilidad dependiendo del lector RFID que se desee usar.

La habilidad de lectura de datos **CRFID\_ReadSkill** posee varios procedimientos o funciones para la correcta comunicación con el lector y lectura de los datos de las etiquetas RFID, éstos son:

- **Constructor de la clase CRFID\_ReadSkill():**

Constructor de la habilidad RFID de lectura. Posee dos parámetros: uno indica el tiempo de espera entre cada ejecución de la habilidad (si es cíclica) y el otro el ID del dispositivo RFID que es habilitado para lectura. El constructor de la habilidad abre el dispositivo USB con ID específico. Se suscribe a los eventos **ACTIVA\_RFID** para activarse como habilidad y **STOP\_RFID** para bloquearse, también es aquí donde se registra el tipo de dato **TAG\_RFID\_DATA** en la memoria compartida a corto plazo.

- **Destructor de la habilidad RFID ~CRFID\_ReadSkill():**

El destructor bloquea la habilidad, cierra el dispositivo RFID, y cancela la suscripción de los eventos **ACTIVA\_RFID** y **STOP\_RFID**.

- **proceso():** Es el proceso principal de la habilidad RFID. El proceso busca etiquetas RFID en el entorno, si encuentra una, lee los datos de la etiqueta y los guarda en la MCP (Algoritmo 1).

---

**Algoritmo 1** Proceso principal Habilidad RFID de lectura

---

```

1: if hay_tag = 0 then
2:   hay_tag = lector.Inventory()           ▷ Busca la etiqueta próxima.
3:
4: else                                     ▷ Si encuentra la etiqueta, lee los datos.
5:   setLectura(read_tag_data(255))
6:
7:   if saveMCDataRFID(getData()) == 0 then
8:                                     ▷ Almacena los datos en la MCP.
9:     hay_tag = 0
10:  end if
11: end if

```

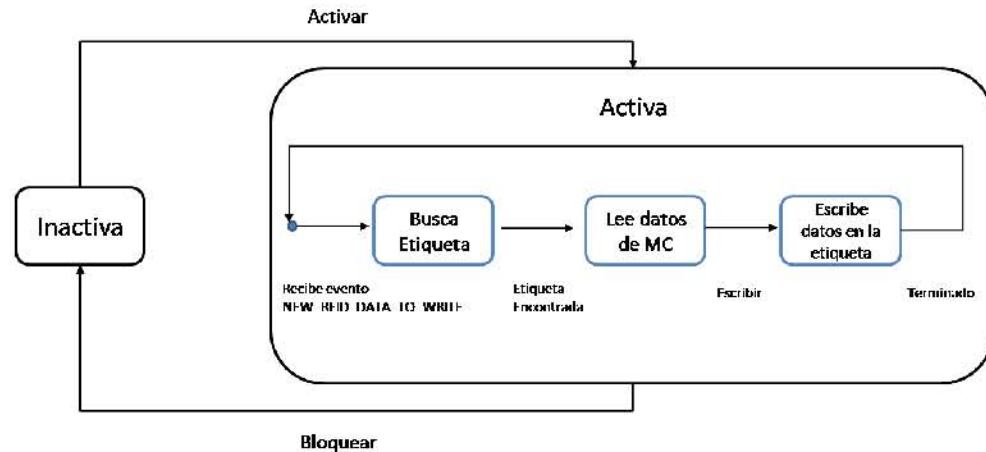
---

- **saveMCDataRFID():** Esta función almacena los datos obtenidos de la etiqueta RFID en la Memoria a Corto Plazo. La función emite el evento **NEW\_TAG\_RFID** indicando que los datos han sido guardados exitosamente y que se encuentran disponibles para las habilidades suscritas.

#### 6.3.4. Habilidad de Escritura de Datos CRFID\_WriteSkill

La habilidad de escritura RFID envía y escribe datos en las etiquetas RFID, se comunica con el lector mediante la interfaz de programación API RFID. Al igual que la habilidad de lectura, es una clase que hereda de la clase base *Habilidad*. Adquiere la información a escribir de la MCP y los envía la etiqueta RFID especificada. El hecho de recibir los datos de la MCP hace el sistema más abierto para que cualquier habilidad o aplicación pueda enviar datos de escritura a las etiquetas. La habilidad es activada cuando recibe un evento indicando que hay nueva información para escribir en las etiquetas. La figura 6.11 muestra el diagrama de estados del proceso de escritura.





**Fig. 6.11:** Diagrama de comportamiento de la habilidad de escritura de datos.

Los atributos de clase de esta habilidad son similares a los de la habilidad de lectura pero con fines distintos:

- **CEventManager emisor:** Es el objeto emisor de eventos. Es usado para manejar la suscripción a los eventos de activación y bloqueo de la habilidad. También gestiona el evento `NEW_RFID_DATA_TO_WRITE` que indica que hay un nuevo dato para escribir en una etiqueta RFID.
- **Tdatos\_etiqueta:** Es el formato en que se guarda la información de las etiquetas en la memoria compartida. Es una estructura con un identificador de dato que indica que es un dato de escritura.
- **Capi\_RFID lector:** Objeto para manejar las funciones básicas de escritura.
- **ID\_Lector:** Número identificador único del lector RFID. Con esto se consigue activar la habilidad dependiendo del lector RFID que se desee usar.

A continuación se describen los principales funciones miembro de la clase `CRFID_WriteSkill`:

■ **Constructor de la clase `CRFID_WriteSkill()`:**

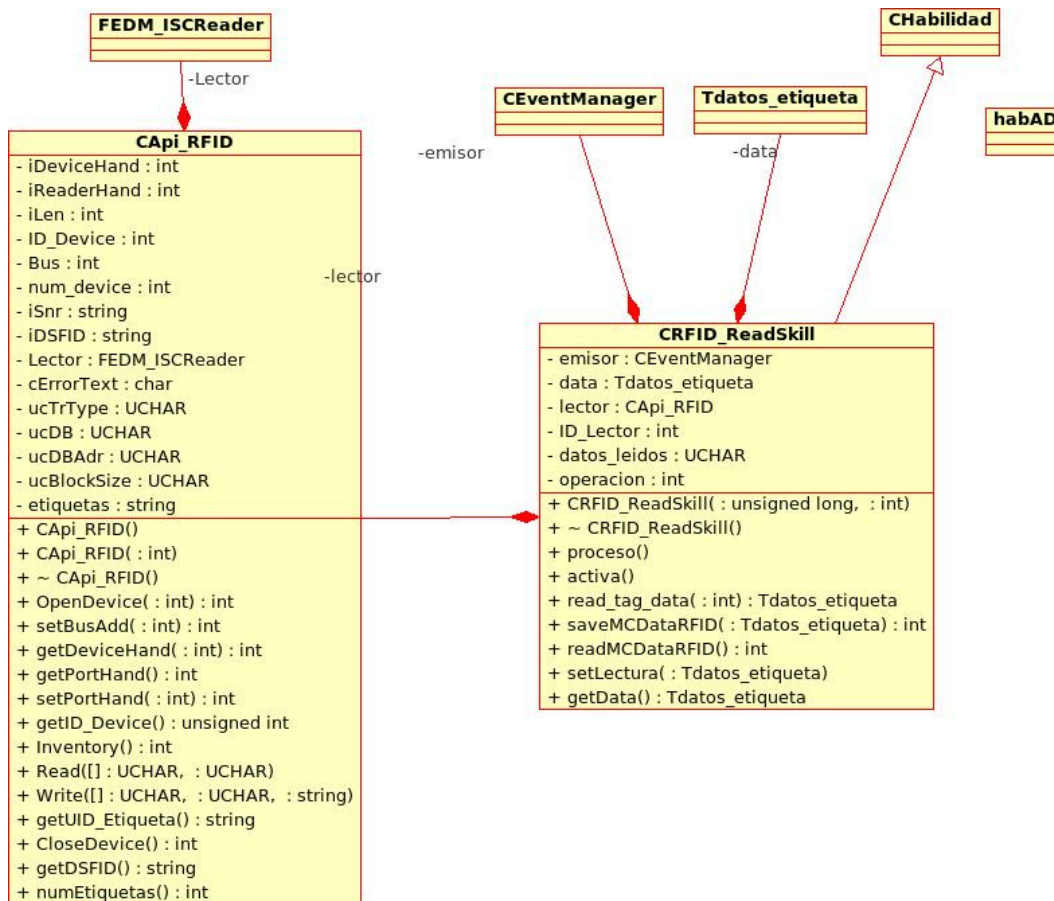
Constructor de la habilidad RFID de escritura. Posee dos parámetros: uno indica el tiempo de espera entre cada ejecución de la habilidad (si es cíclica) y el otro el ID del dispositivo RFID que es habilitado para escritura. El constructor de la habilidad abre el dispositivo USB con ID específico. Se suscribe a los eventos `ACTIVA_RFID` para activarse como habilidad y `STOP_RFID` para bloquearse. Para saber cuando hay un dato nuevo para escribir en una etiqueta específica, se suscribe al evento `NEW_RFID_DATA_TO_WRITE`.

■ **Destructor de la habilidad RFID `~CRFID_WriteSkill()`:**

El destructor bloquea la habilidad, cierra el dispositivo RFID, y cancela la suscripción de los eventos `ACTIVA_RFID`, `STOP_RFID` y `NEW_RFID_DATA_TO_WRITE`.

- **proceso()**: Es el proceso principal de la habilidad RFID. Se encarga de buscar constantemente etiquetas RFID en el entorno.
- **escribir\_datos()**: Es la función que se activa cuando es recibido el evento NEW\_RFID\_DATA\_TO\_WRITE. Lee la información que se desea escribir en una etiqueta RFID desde la MCP.

En las figuras 6.12 y 6.13 se muestran los diagramas de clase de las habilidades RFID desarrolladas.

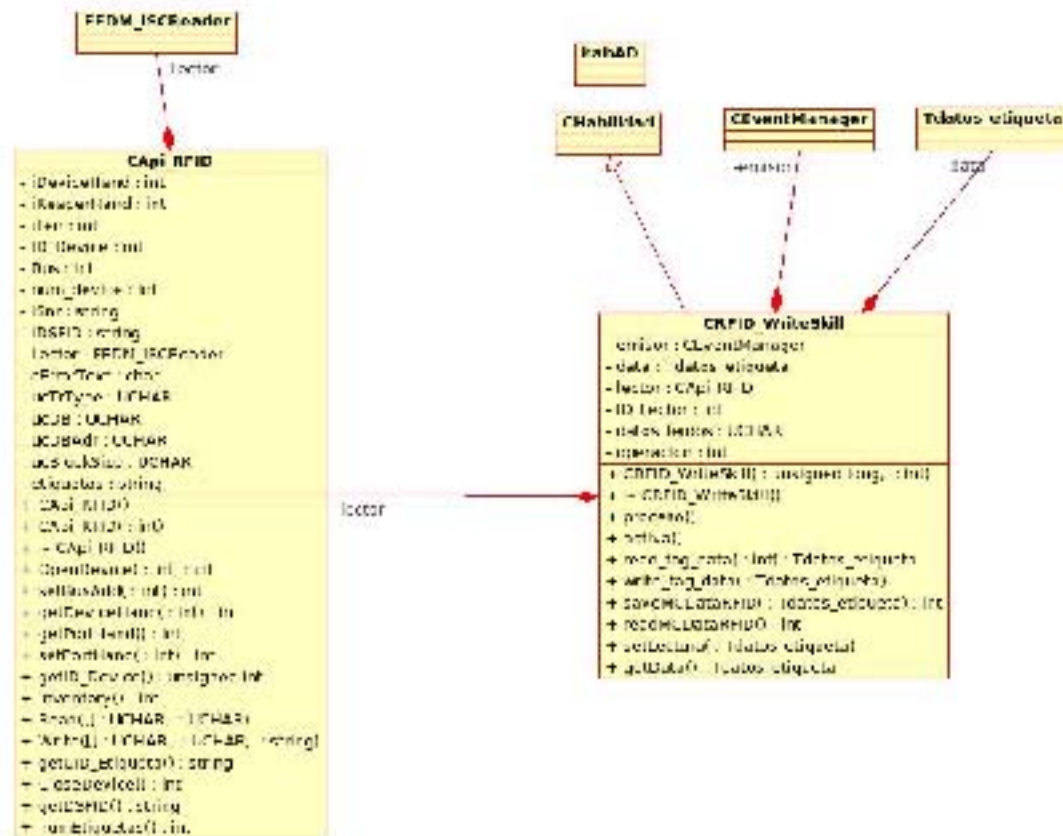


**Fig. 6.12:** Diagrama de Clases de la Habilidad de Lectura de Etiquetas RFID.

## 6.4. Información en las etiquetas RFID

Las etiquetas RFID almacenarán la siguiente información::

- **UID:** Es un identificador único de la etiqueta asignado por el fabricante.
- Un identificador que representa el tipo de información que guarda la etiqueta: información para navegación o información útil para reconocimiento de objetos.



**Fig. 6.13:** Diagrama de Clases de la Habilidad de Escritura de Etiquetas RFID.

- La información de la señal: lugar, conexiones y acciones.

Las etiquetas seleccionadas tienen una capacidad máxima de almacenamiento de 1024 bytes para los lectores HF 13,5MHz y para los lectores UHF, las etiquetas pasivas seleccionadas tienen una capacidad máxima de almacenamiento de 12 bytes . Debido a la limitación de almacenamiento en la tecnología actual de etiquetas RFID, no es posible almacenar toda la información de cada señal del sistema de navegación. Es por eso que se ha optado almacenar la dirección de un fichero XML que guarda la información de la señal (lugar, conexiones y acciones).

Guardar físicamente toda la información del entorno que se ha modelado en las etiquetas, actualmente, no es conveniente por la siguientes razones:

1. Espacio físico: La cantidad de bytes disponibles en las etiquetas es escaso y no es posible guardar toda la información.
2. La diferencia de tamaño de la memoria física disponible en las diferentes etiquetas RFID existentes en el mercado.

Para la representación del conocimiento del entorno se opta por usar XML ya que:

- Ofrece la posibilidad de representar la información de forma neutra, independiente del lenguaje de programación y del sistema operativo empleado.

- Es extensible: en cualquier momento podemos añadir nuevos elementos XML, de modo que se pueda continuar utilizando sin modificaciones complejas.
- Su estructura es sencilla de comprender y procesar.
- Permite que cada elemento represente un concepto o entidad a la cual se puede asociar un contexto específico.

La estructura básica de una señal la modelamos entonces con un documento XML con la siguientes etiquetas:

**Lugar** Un lugar se identificará con el elemento `< place >`, éste será el elemento principal y contendrá las diferentes zonas que conforman el lugar.

Cada zona posee un nombre y un identificador alfanumérico. La definición en XML de un lugar entoves sería de de la siguiente manera:

```
<place>
  <zone>
    <label>Nombre de la zona 1</label>
    <ID>Identificador zona 1</ID>
  </zone>

  ...

  <zone>
    <label>Nombre de la zona n</label>
    <ID>Identificador zona n</ID>
  </zone>
</place>
```

**Conexión** Como se ha dicho anteriormente una conexión está conformada por un lugar y una acción. En XML se usará la siguiente estructura para definirla:

```
<connection>
  <place>
    <zone>
      <label>Nombre de la zona 1</label>
      <ID>Identificador zona 1</ID>
    </zone>

    ...

    <zone>
      <label>Nombre de la zona n</label>
      <ID>Identificador zona n</ID>
    </zone>
  </place>
```

```

    <action>
      <nombre>Acción que debe realizar el robot</nombre>
      <tipo>int</tipo>
      <codigo>int</codigo>
    </action>
  </connection>

```

Una señal entonces estará conformada por:

- Un lugar, que indica el lugar actual que indica la señal.
- Una conexión o lista de de conexiones, desde donde se puede ir a partir del lugar que indica la señal

La estructura básica será entonces de la siguiente manera:

```

<place>
  <zone>
    <label>Nombre de la zona 1</label>
    <ID>Identificador zona 1</ID>
  </zone>

  ...

  <zone>
    <label>Nombre de la zona n</label>
    <ID>Identificador zona n</ID>
  </zone>
</place>

<connection>
  <place>
    <zone>
      <label>Nombre de la zona A hacia donde se puede ir</label>
      <ID>Identificador zona A </ID>
    </zone>

    ...

  </place>
  <action>
    <nombre>Acción que debe realizar el robot para ir a la zona A</nombre>
    <tipo>int</tipo>
    <codigo>int</codigo>
  </action>
</connection>

<connection>
  <place>
    <zone>
      <label>Nombre de la zona B hacia donde se puede ir</label>
      <ID>Identificador zona B </ID>
    </zone>
  </place>
  <action>
    <nombre>Acción que debe realizar el robot para ir a la zona B</nombre>
    <tipo>int</tipo>
    <codigo>int</codigo>
  </action>
</connection>

```

```

    </zone>

    ...

</place>

<action>
    <nombre>Acción que debe realizar el robot para ir a la zona B</nombre>
    <tipo>int</tipo>
    <codigo>int</codigo>
</action>
</connection>

...

<connection>
    <place>
        <zone>
            <label>Nombre de la zona K hacia donde se puede ir</label>
            <ID>Identificador zona K </ID>
        </zone>

        ...

    </place>
    <action>
        <nombre>Acción que debe realizar el robot para ir a la zona K</nombre>
        <tipo>int</tipo>
        <codigo>int</codigo>
    </action>
</connection>

```

## 6.5. Estructura Software de las Señales

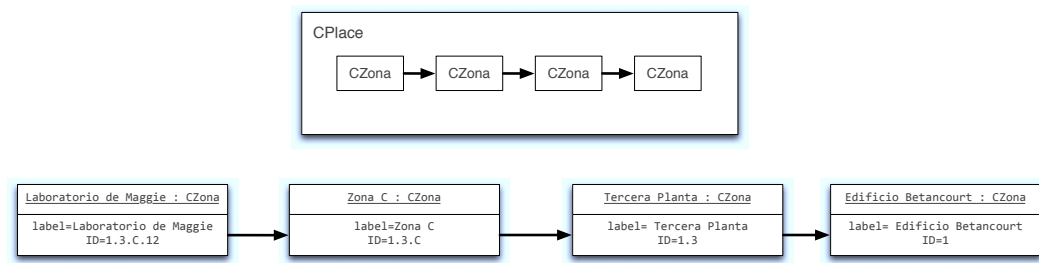
En esta sección explicaremos los diferentes elementos software para la implementación del algoritmo de navegación presentado en esta tesis. El enfoque utilizado a nivel de software es el diseño orientado a objetos, ya que, además de los diferentes beneficios que tiene este paradigma de diseño, como la reutilización del software, facilita la integración en la arquitectura de software del robot Maggie en el que se desarrollará esta tesis.

En términos de software una señal será un objeto denominado *Nodo de Navegación* (CNavigation\_Node), es el objeto que contiene la información almacenada en la etiqueta RFID. Corresponde a la señalización que el robot lee del entorno. Está conformado por 4 componentes:

1. Lugar (CPlace): Es el objeto que define un *lugar*. Como se dijo anteriormente, un lugar está conformado por un conjunto de *zonas* anidadas. Un objeto CZona posee dos elementos: nombre e identificador. A nivel de software, representaremos un *lugar* como una *lista de zonas*, ordenadas desde la mas específica a la

mas genérica. La figura 6.14, muestra la representación del lugar: *Laboratorio de Maggie*, a través de una lista de zonas:

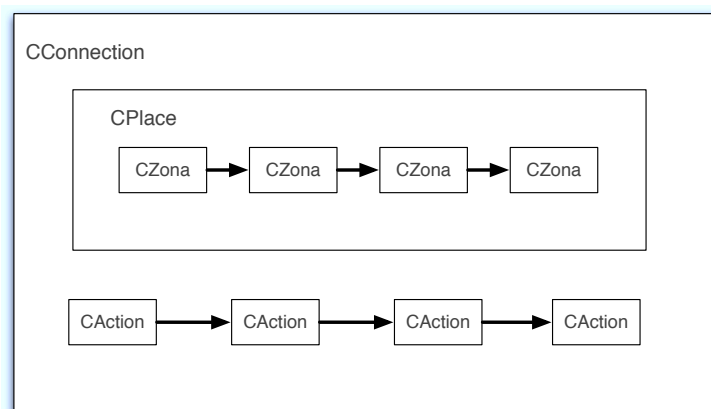
- Laboratorio de Maggie – 1.3.C.12
- Zona C – 1.3.C
- Tercera Planta – 1.3
- Edif. Betancourt – 1



**Fig. 6.14:** Representación de un lugar: *Laboratorio de Maggie*.

2. Lista de Conexiones: Un *nodo de navegación* contiene una lista de conexiones que indican los lugares a donde se puede ir desde el lugar en el que se ha detectado la etiqueta RFID.

Un objeto *conexión* (**CConnection**) hereda de la clase **CPlace**, posee una lista de acciones (**CAction**), que debe ejecutar el robot para llegar al lugar que indica la conexión. La figura 6.15 muestra el esquema de una *conexión*.



**Fig. 6.15:** Representación de una conexión.

Una acción tiene dos atributos básicos:

- Nombre: define con un lenguaje natural lo que el robot debe ejecutar.

- Tipo: indica el tipo de acción, bien sea topológica o geométrica.  
Si el tipo de acción es geométrica una acción debe poseer dos elementos adicionales:
  - Mapa: La dirección del fichero del mapa geométrico del entorno.
  - Ubicación: Esto es, la localización con respecto al mapa geométrico en coordenadas  $(x, y, \theta)$ .

## 6.6. Habilidad de Navegación Mediante Señales

El algoritmo de navegación diseñado se va a implementar e incorporar en el robot como una habilidad. Este proceso se realizará en 2 fases:

1. Proceso de lectura de las señales del entorno.
2. Procesamiento de la información obtenida.

### 6.6.1. Lectura de las señales

Como se ha dicho previamente, las señales que vamos a usar en este sistema son etiquetas RFID. Para leer las etiquetas se ha incorporado una antena RFID y previamente se ha desarrollado el software para la integración de la misma en el robot, además de una habilidad de lectura de etiquetas, integrada en la arquitectura de software y que se ha implementado en el robot Maggie del RoboticsLab.

La habilidad de lectura RFID se mantiene en ejecución constante detectando las etiquetas que se encuentran en su entorno inmediato, ésto permite que en cualquier momento de la navegación el robot pueda leer una señal e interpretarla.

Los usuarios principales de la habilidad de lectura RFID son otras habilidades del sistema. Éstas se suscriben al evento `NEW_TAG_RFID`, que indica que los datos leídos por el lector RFID, se encuentran en la Memoria a Corto Plazo del sistema.

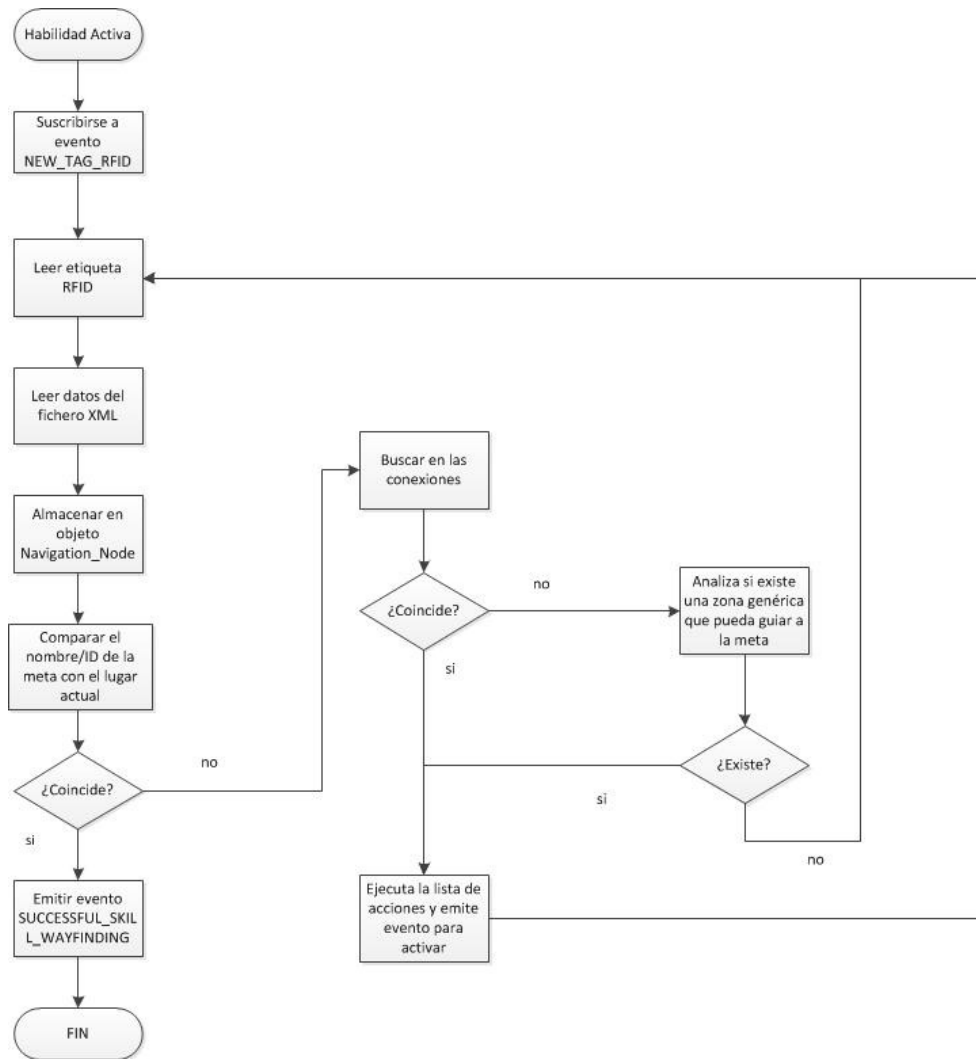
### 6.6.2. Procesamiento de la información

La figura 6.16, muestra el diagrama de flujo de la *Habilidad de Navegación Mediante Señales*. Cuando la habilidad es activada por otra habilidad o un secuenciador de habilidades, se suscribe al evento `NEW_TAG_RFID`, evento que emite la habilidad de lectura RFID (visto en la sección 6.3.2). La habilidad lee la información almacenada en la etiqueta RFID y analiza esta información. Lee e interpreta los datos y los guarda en un objeto `CNavigation_Node`.

Una vez leído, comienza a hacer la comparación del nombre del lugar y el identificador de la meta asignada con los datos del nodo de navegación. La comparación maneja 3 casos posibles y es realizada según el siguiente orden:

**Caso 1:** Compara el nombre y/o el identificador de la meta asignada con el lugar actual que indica la etiqueta. Si coincide, la habilidad emite el evento `SUCCESSFUL_SKILL_WAYFINDING`, que indica que el robot ha llegado satisfactoriamente a la





**Fig. 6.16:** Diagrama de flujo de la Habilidad de Navegación Mediante Señales.

meta. Una vez emitido, la habilidad pasa al estado *bloqueada* en espera de una nueva orden.

**Caso 2:** En caso de no coincidir la meta con el lugar actual, procede a analizar las conexiones. Evalúa los lugares que indica la lista de conexiones y compara el nombre del lugar y/o el identificador. En caso de coincidir, ejecuta la acción o lista de acciones que señala.

**Ejemplo:** Si la meta asignada es: *Laboratorio de Maggie – 1.3.C.12*

y las conexiones del lugar que indica la etiqueta son:

- Servicios –1.3.C.S – Avanzar por la izquierda.
- Laboratorio de Robótica III – 1.3.C.14 – Avanzar por la derecha.

- Laboratorio de Maggie –1.3.C.12 – Avanzar por la izquierda.

La habilidad encuentra una conexión que indica como llevarle a la meta, entonces la habilidad desactiva las habilidades de movimiento del robot que haya activado previamente para la búsqueda de la etiqueta RFID o bien para seguir una indicación. Luego ejecuta la acción que señala la conexión: *Avanzar por la izquierda*, el manejador de acciones de la habilidad emite un evento para activar la habilidad “*Seguir contorno por la izquierda*”.

**Caso 3:** Si no se cumplen el caso 1 y 2, la habilidad busca en las conexiones una zona genérica que pueda orientarlo y llevarlo hacia la meta.

**Ejemplo:** Si la meta asignada es: *Laboratorio de Maggie – 1.3.C.12*, el lugar que indica la etiqueta es: *Departamento de Ingeniería de Organización – 1.3.A* y las conexiones son:

- Departamento de Sistemas y Automática Zona B –1.3.B – Avanzar por la izquierda
- Departamento de Sistemas y Automática Zona C – 1.3.C – Avanzar por la izquierda.
- Departamento de Ingeniería Electrónica – 1.3.D – Avanzar por la izquierda.

Al no encontrar el identificador específico *1.3.C.12*, la habilidad comenzará a descomponer el identificador para buscar la zona donde se encuentra la meta.

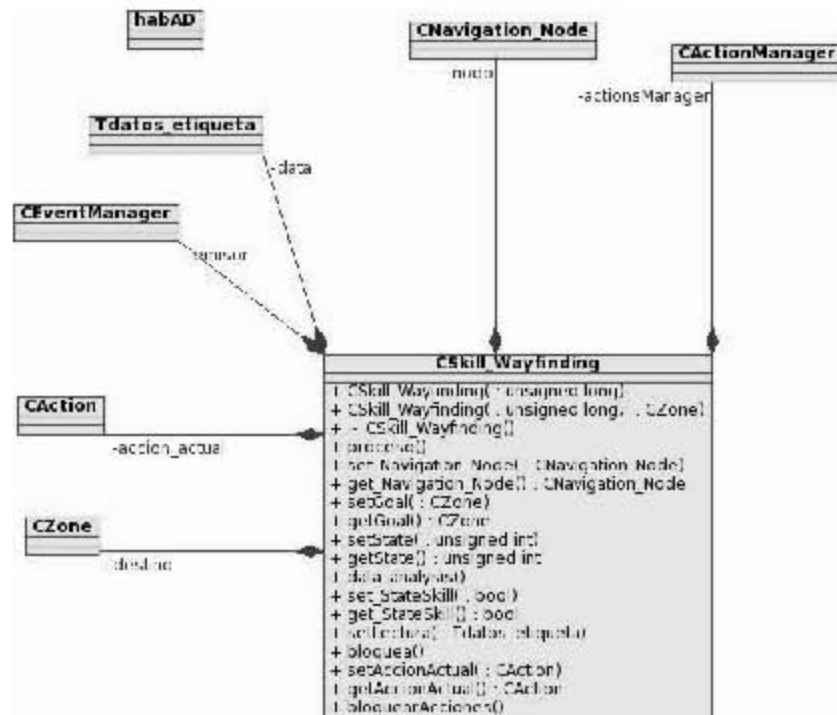
Para el ejemplo, el robot seguirá las instrucciones para ir a las zona 1.3.C y ejecutará la acción *Avanzar por la izquierda*, entonces el manejador de acciones de la habilidad emitirá un evento para activar la habilidad “*Seguir contorno por la izquierda*”.

Si no se cumple ninguno de los casos anteriores el robot seguirá explorando para buscar una nueva señal y seguir con el proceso de orientación y búsqueda de señales.

La figura 6.17 muestra el diagrama de clases de la Habilidad de Navegación (CSkill\_Wayfinding) y sus componentes.

Las características más importantes de la habilidad de navegación son las siguientes:

1. Hereda de la clase abstracta *habilidad* [habAD], ya que debe formar parte de la arquitectura AD implementada en el robot Maggie.
2. Tiene un manejador de eventos [CEventManager] para recibir y enviar notificaciones desde y hacia otras habilidades del sistema. Recibe la notificación de detección y lectura de datos ([Tdatos\_etiqueta]) de las etiquetas RFID desde las habilidades básicas de RFID (suscribiéndose al evento NEW\_TAG\_RFID) y emite el evento SUCCESSFUL\_SKILL\_WAYFINDING para notificar a otras habilidades que ya ha alcanzado la meta con éxito.



**Fig. 6.17:** Diagrama de clases de la Habilidad de Navegación Mediante Señales.

3. Usa los objetos Nodo de Navegación y sus componentes: Lugar [CPlace], Zona [CZone] y Action [CAction].
4. Usa una clase Manejadora de Acciones [CAction\_Manager] que se encarga de la gestión para la activación y bloqueo de las habilidades que indican las acciones en la señalización hacia la meta. A esta clase se le asigna una lista de eventos que se emitirán dependiendo de las indicaciones de las señales, por ejemplo, si la señal indica que debe ejecutar la acción "Avanza hacia la derecha", el Manejador de acciones desactivará las habilidades relacionadas con la navegación (específicamente las de movimiento del robot) y emitirá la activación de la habilidad "Seguir el Contorno por la derecha".

El sistema de navegación contempla la posibilidad de usar mapas geométricos a nivel local en el caso de entornos no estructurados (como habitaciones amplias), aquí el Manejador de Acciones, gestionará habilidades geométricas que usen la información del mapa geométrico.

En el caso que el tipo de acción a ejecutar sea geométrica, se debe obtener la dirección del fichero donde está almacenado el mapa geométrico. Esta dirección es un atributo del objeto [CNavigation\_Node]. En el siguiente ejemplo de la estructura de contenido de una señal, los datos del mapa se encuentran contenidos en el elemento <GeometricMap>.

```
<connection>
  <place>
```

```

    <zone>
      <label> TV </label>
      <ID> 1.3.C.12.TV </ID>
    </zone>
  </place>
  <action>
    <nombre>GTP (-1.15,0,0)</nombre>
    <tipo>1</tipo>
    <codigo>3</codigo>
    <GeometricMap>
      <direccion>file:///home/ana/Wayfinding/mapa\_lab.xml</direccion>
      <localizacion>(0.25,0.25,0)</localizacion>
    </GeometricMap>
  </action>
</connection>

```

El fichero donde está almacenado el mapa está en formato XML y se encuentra identificado por el elemento `<direccion>`, allí se almacena la dirección URL de la imagen que representa el mapa geométrico (en formato PNG), las dimensiones, la resolución y su descripción. También es posible indicar la localización del robot con respecto al mapa geométrico, identificado por el elemento `<localizacion>`.

A continuación se muestra el contenido del fichero `mapa_lab.xml` :

```

<mapa>
  <geometricMap>
    <image>
      <title>Mapa del laboratorio 1.3.C12</title>
      <url>file:///home/corrales/mapa_3cm_centrado_bn_fit.png</url>
      <width>229</width>
      <height>404</height>
      <tipo>1</tipo>
      <resolution>0.03</resolution>
      <description>Mapa 1.3.C12</description>
    </image>
  </geometricMap>
</mapa>

```

El Manejador de Acciones al cargar el mapa geométrico activa la habilidad de localización `CSkill_Geometric_Localize`, es una habilidad que se encarga de localizar geométricamente al robot usando el algoritmo AMCL (Adaptive Monte-Carlo Localization algorithm) de *Player* (ver C.1), para este fin, usa los parámetros de información geométrica dada por el mapa y por el objeto `CNavigation_Node`.

La figura 6.18 muestra el mapa geométrico del laboratorio 1.3.C.12, realizado con la herramienta *pmap*.



*Fig. 6.18: Mapa geométrico del laboratorio 1.3.C.12.*

## 6.7. Diálogos para la adquisición de información en la señalización

Como se ha dicho anteriormente, la gestión de diálogos dinámicos está basado en el estándar Voice eXtensible Markup Language (*voiceXML*).

Para el diseño de la aplicación de diálogos VoiceXML, es muy importante analizar y estudiar los contenidos de cada diálogo, es decir, la información que el robot recibirá del usuario y la que el usuario recibirá del robot, así como el flujo de control de la información.

El conjunto de diálogos funciona como una máquina de estados, en la que en cada diálogo se especifica la información que el robot enviará al usuario, la información que proporcionará éste y, dependiendo de las respuestas recibidas o enviadas, se pasará al siguiente estado.

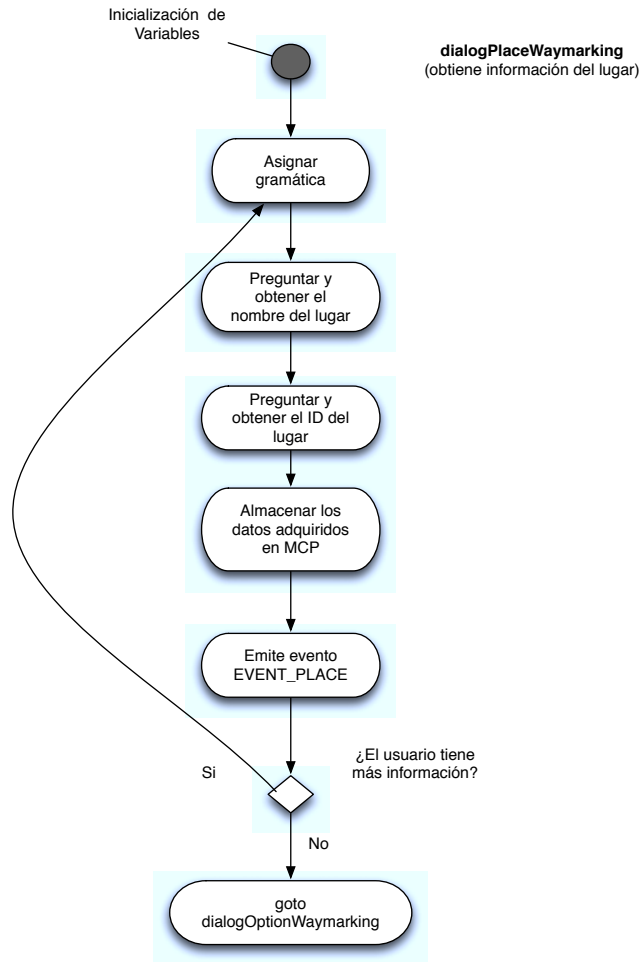
En un diálogo VXML, se usan formularios `<form>` para adquirir información del usuario (similares a los formularios web). Estos formularios están compuestos por diferentes campos, relacionados directamente con la información que el robot requiere para escribir una señal (variables semánticas).

Para especificar los mensajes que el robot envía al usuario, se utiliza la etiqueta `<prompt>`. El texto contenido en ésta etiqueta será sintetizado por el robot usando la habilidad de síntesis de habla *ettsSkill*.

A continuación se muestran 3 diagramas de actividad que modelan los aspectos dinámicos de la estructura de los diálogos básicos para la interacción con el robot, mostrados anteriormente en 5.2.1. El ejemplo que se muestra corresponde a el caso en que el robot toma la iniciativa en el diálogo. En éstos se puede observar las diferentes

fases del diálogo para obtener la información de una señal.

### 6.7.1. Parte 1: dialogPlaceWaymarking



**Fig. 6.19:** Diagrama de actividades: *dialogPlaceWaymarking*.

En la figura 6.19, se presenta el esquema de funcionamiento del primer diálogo *dialogPlaceWaymarking*. Éste diálogo obtiene del usuario información del lugar actual (es decir donde se encuentra la señal). Primeramente en el *vxm1* se inicializan el conjunto de variables que se van a usar para almacenar los datos y las variables que representan a los eventos que notificarán al sistema la disponibilidad de nuevos datos en la Memoria a Corto Plazo. Luego se asigna la gramática que se va a utilizar, que contendrá los elementos semánticos del lenguaje (p.ej. la vista en 5.3.1). Una vez hecho ésto se procede a ejecutar el formulario. El robot comienza a solicitar información al usuario sobre las características del **lugar** donde está ubicada la señal, el nombre del lugar y su ID. Al recibir los datos almacena los datos obtenidos en la Memoria a Corto Plazo y notifica al sistema emitiendo un evento (*EVENT\_PLACE*). Posteriormente el robot pregunta al usuario si tiene más información de ese lugar, si es así comienza de

nuevo el diálogo, éste es con el objetivo de completar la información en el caso de que el lugar esté compuesto por *zonas anidadas*. En el caso de que el usuario no posea más información el sistema pasa al siguiente diálogo (`dialogOptionWaymarking`).

A continuación se presenta la estructura inicial en VXML donde se declaran las variables y los eventos para el desarrollo de los diálogos :

- Inicialmente se declaran variables que usará el intérprete VXML para almacenar información, como el nombre del lugar, identificador, nombre de conexiones y sus identificadores, acciones y tipo de acciones (líneas 6 a 18).
- También se hace la declaración de variables que representan los eventos que se emitirán durante la ejecución del diálogo. Éstos eventos indicarán al sistema cuando se han almacenado en la Memoria a Corto Plazo del sistema los datos de la señal, para que puedan ser utilizados por otras habilidades, en este caso, la habilidad que escribirá los datos en las etiquetas de RFID (líneas 19 a 24).

#### *Declaración e inicialización de variables*

```

1 <vxml xsi:schemaLocation="http://www.w3.org/2001/vxml
2                               http://www.w3.org/TR/voicexml20/vxml.xsd"
3                               version="2.0">
4
5 <property name="confidencelevel" value="0.45"/>
6 <var name="i"/>
7 <var name="place_label"/>
8 <var name="place_ident"/>
9 <var name="info" expr="1"/>
10 <var name="connection_id"/>
11 <var name="connection_label"/>
12 <var name="action_label"/>
13 <var name="code_action"/>
14 <var name="tipo_action"/>
15 <var name="info_c" expr="1"/>
16 <var name="info_a" expr="1"/>
17 <var name="indicaciones"/>
18 <var name="pregunta1"/>
19 <var name="EVENT_PLACE"/>
20 <var name="EVENT_CONNECTION"/>
21 <var name="EVENT_ACTION"/>
22 <var name="EVENT_SAVE" expr="500107"/>
23 <var name="EVENT_ENDING_CONNECTION"/>
24 <var name="EVENT_NO_ENDING_CONNECTION"/>

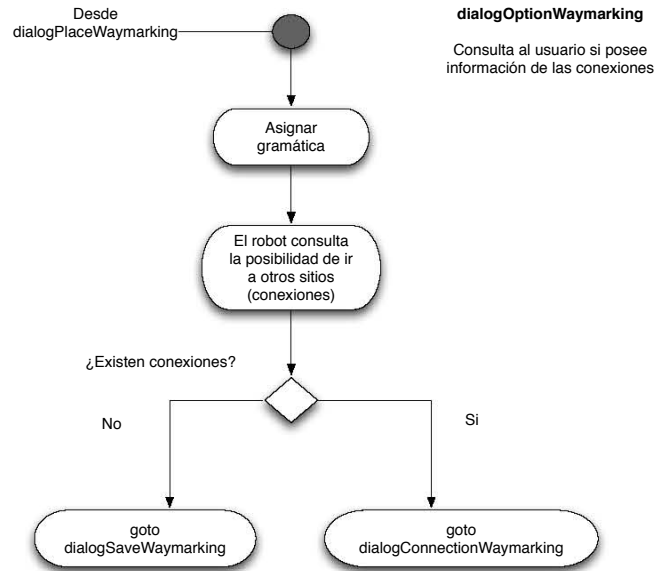
```

La estructura del diálogo `dialogPlaceWaymarking`, se muestra en la sección 7.3.2 de los Resultados Experimentales.

### 6.7.2. Parte 2: `dialogOptionWaymarking`

Una vez que se haya adquirido la información del lugar, el robot procede a preguntar por la existencia de conexiones. Si el usuario no confirma la posibilidad de ir a otros

sitios, el sistema termina el diálogo y llama al formulario `dialogSaveWaymarking`, de lo contrario, se pasa al siguiente diálogo, encargado de obtener la información de las conexiones (`dialogConnectionWaymarking`). En la figura 6.20, se muestra el diagrama de actividades del diálogo `dialogOptionWaymarking`. La implementación en VXML se encuentra en la sección 7.3.2 de los Resultados Experimentales.



**Fig. 6.20:** Diagrama de actividades: *dialogOptionWaymarking*.

### 6.7.3. Parte 3: `dialogConnectionWaymarking`

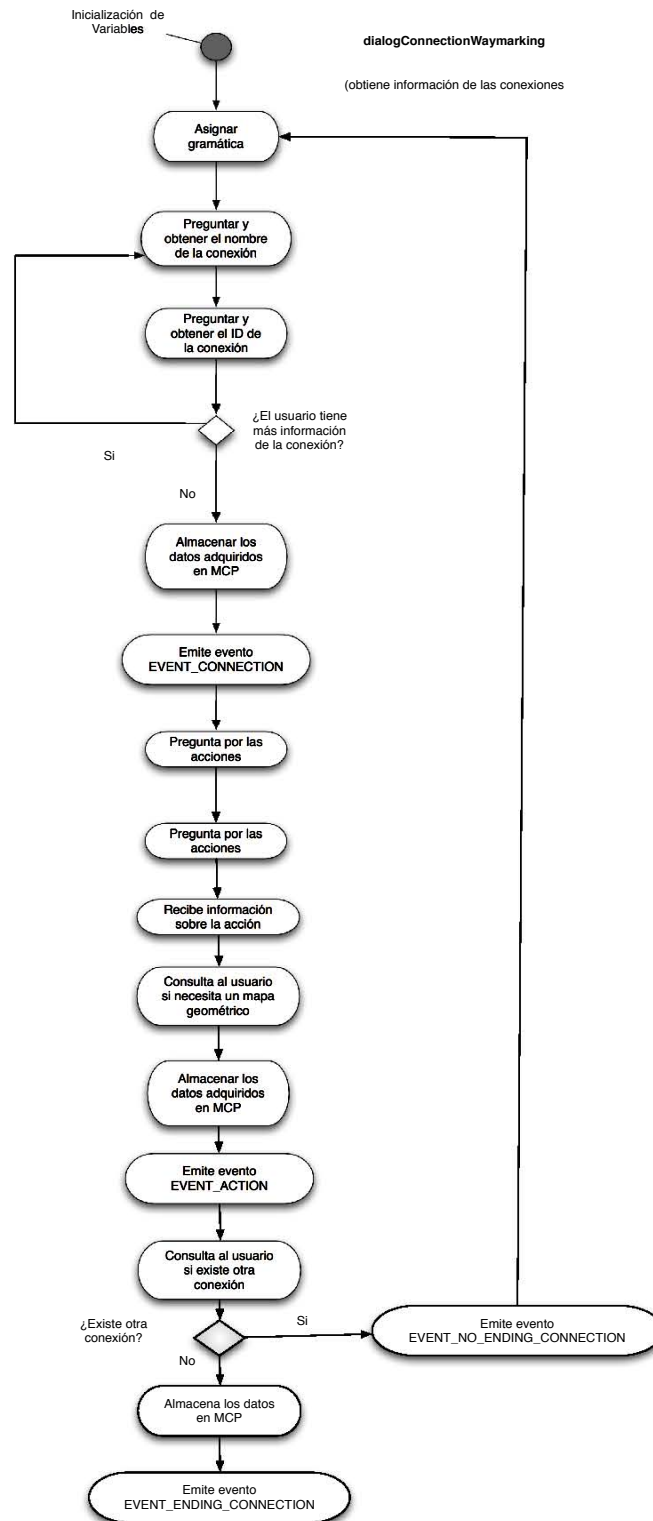
Si el usuario indica al robot que posee información sobre otros lugares a donde es posible ir desde el lugar donde se encuentra la señal, se ejecuta el diálogo `dialogConnectionWaymarking`. En la figura 6.21 se presenta el diagrama de actividades de éste diálogo.

Inicialmente se asigna la gramática que se va a usar. Posteriormente el robot procede a consultar los datos de cada conexión, pregunta por el nombre y por el ID de la conexión. Las preguntas son simples, similares a las vistas en el diálogo de adquisición de información de un lugar.

Como las conexiones representan lugares, es posible que pueda contener información anidada (zonas anidadas), así que el robot preguntará hasta que el usuario confirme que no tiene más información de la conexión en particular. Cuando el robot haya adquirido la información, el gestor de diálogos almacenará la información en la Memoria a Corto Plazo del sistema y emitirá el evento `EVENT_CONNECTION`.

La siguiente parte del diálogo complementa la información de la conexión. El robot pregunta al usuario por la acción que se tiene que realizar para ir al lugar indicado. Como el tipo de acción puede ser geométrica, consulta al usuario si es necesario un mapa geométrico. Una vez que haya completado la información de la acción, el gestor



Fig. 6.21: Diagrama de actividades: `dialogConnectionWaymarking`.

de diálogo almacena la información en la Memoria a Corto plazo y emite un evento para notificar a las habilidades o procesos interesados en los datos.

El sistema repite el proceso si el usuario comunica la existencia de otras conexiones, si el usuario no posee más información, el gestor de diálogos notifica al sistema que ha recibido correctamente la información de las conexiones para el lugar donde se encuentra la señal a escribir.

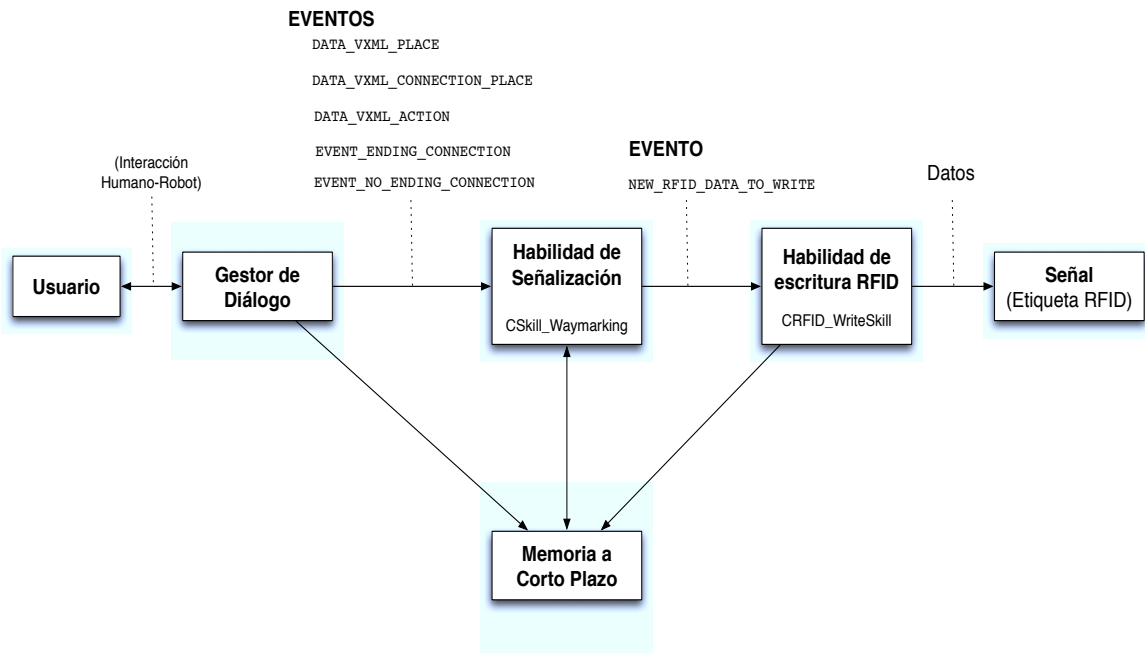
La estructura en VXML del diálogo `dialogConnectionWaymarking` se presenta en la sección 7.3.2 de los Resultados Experimentales.

## 6.8. Habilidad de Señalización

La Habilidad de Señalización (`CSkill_Waymarking`), es la habilidad desarrollada e integrada en el sistema de navegación presentado en ésta tesis, cuyo objetivo es gestionar la información adquirida a través de la interacción humano-robot y darle el formato adecuado de una señal, para posteriormente enviarla a la habilidad de escritura de señales.

La habilidad de señalización estará atenta a los eventos emitidos por el Gestor de Diálogo y leerá de la Memoria Corto Plazo del sistema, los datos del lugar y la conexión que hayan sido proporcionados por el usuario.

En la figura 6.22 se muestra el esquema del sistema del señalización con la habilidad `CSkill_Waymarking`.



**Fig. 6.22:** Esquema del sistema de señalización y sus diferentes elementos.

Como se puede observar en la figura, la habilidad de señalización sirve de intermediaria entre el Gestor de Diálogo y la habilidad de escritura `CRFID_WriteSkill`.

Está suscrita a los diferentes eventos que emite el Gestor de Diálogo, éstos eventos notifican a la habilidad que han sido almacenados, en la Memoria a Corto Plazo del sistema, los datos de una señal. Los eventos son:

- **DATA\_VXML\_PLACE**: Notifica a la habilidad que ha sido almacenada información de un lugar. Este evento es enviado cada vez que el Gestor de Diálogo almacena información de una zona específica de un lugar. Si el lugar  $l_i$  está conformado por  $n$  zonas, de modo que:  $l_i = \{z_1, \dots, z_n\}$ , entonces éste evento es emitido al recibir información de cada zona  $z_i$ .

Al recibir el evento, la habilidad lee de la Memoria a Corto Plazo la siguiente información:

- *label*: Corresponde al nombre de la zona.
  - *ID*: Identificador alfanumérico de la zona.
  - *option*: Variable para determinar si se ha obtenido la información de las  $n$  zonas del correspondiente lugar  $l_i$ .
- **DATA\_VXML\_CONNECTION\_PLACE**: Notifica a la habilidad que ha sido almacenada información de una conexión. Como se ha visto en la sección 4.2, una conexión está conformada por un lugar  $l_c$  y una acción  $a_c$ . Este evento entonces, notificará la información del lugar  $l_c$  que indica la conexión. Los datos recibidos corresponden al esquema de un lugar:
  - *label*: Corresponde al nombre de la zona que indica la conexión.
  - *ID*: Identificador alfanumérico de la zona.
  - *option*: Variable para determinar si se ha obtenido la información de las  $n$  zonas del correspondiente lugar  $l_i$  que indica la conexión.
- **DATA\_VXML\_ACTION**: Notifica a la habilidad que ha sido almacenada información de una acción. Los datos complementan la información de una conexión  $c_i$ . Indica la acción o conjunto de acciones que se deben realizar para llegar al lugar  $l_i$ , que ha sido previamente notificado con el evento **DATA\_VXML\_CONNECTION\_PLACE**. La habilidad de señalización al recibir el evento **DATA\_VXML\_ACTION**, la habilidad lee de la Memoria a Corto Plazo la siguiente información:
  - *label*: Nombre o descripción de la acción.
  - *type*: Tipo de acción. Ésto es si es topológica o geométrica.
- **EVENT\_ENDING\_CONNECTION** y **EVENT\_NO\_ENDING\_CONNECTION**: Estos eventos permiten a la habilidad saber si han sido almacenados en la Memoria a Corto Plazo la totalidad de las conexiones de un lugar. Es decir, el sistema necesita saber cuando se ha terminado de enviar la información de las conexiones. El Gestor de Diálogo lo notifica cuando el usuario le dice que no conoce o no puede indicarle otros lugares hacia donde puede ir desde el lugar donde el robot se encuentre escribiendo una señal.

Una vez que el proceso de la habilidad da formato a la información recibida desde el gestor de diálogo, almacena los datos de la señal en la Memoria a Corto Plazo del sistema y emite el evento `NEW_RFID_DATA_TO_WRITE`. A éste evento está suscrito la habilidad de escritura de etiquetas RFID, encargada de escribir físicamente la señal.

### 6.8.1. Implementación de la Habilidad de Señalización

En la figura 6.23 se muestra el diagrama de comportamiento de la *Habilidad de Señalización*. Cuando la habilidad es activada por otra habilidad o un secuenciador de habilidades, se suscribe a los eventos explicados anteriormente, emitidos por el Gestor de Diálogo.

Dependiendo del evento recibido, la habilidad tendrá un comportamiento diferente:

- Si recibe el evento `DATA_VXML_PLACE`, la habilidad leerá datos de un *lugar* en la Memoria a Corto Plazo.
- Si recibe el evento `DATA_VXML_CONNECTION_PLACE`, la habilidad leerá, en la Memoria a Corto Plazo, datos parciales de una *conexión* (el lugar a donde es posible ir).
- Si recibe el evento `DATA_VXML_ACTION`, la habilidad leerá, de la Memoria a Corto Plazo, la *acción* que se debe realizar para ir al lugar indicado en la conexión.

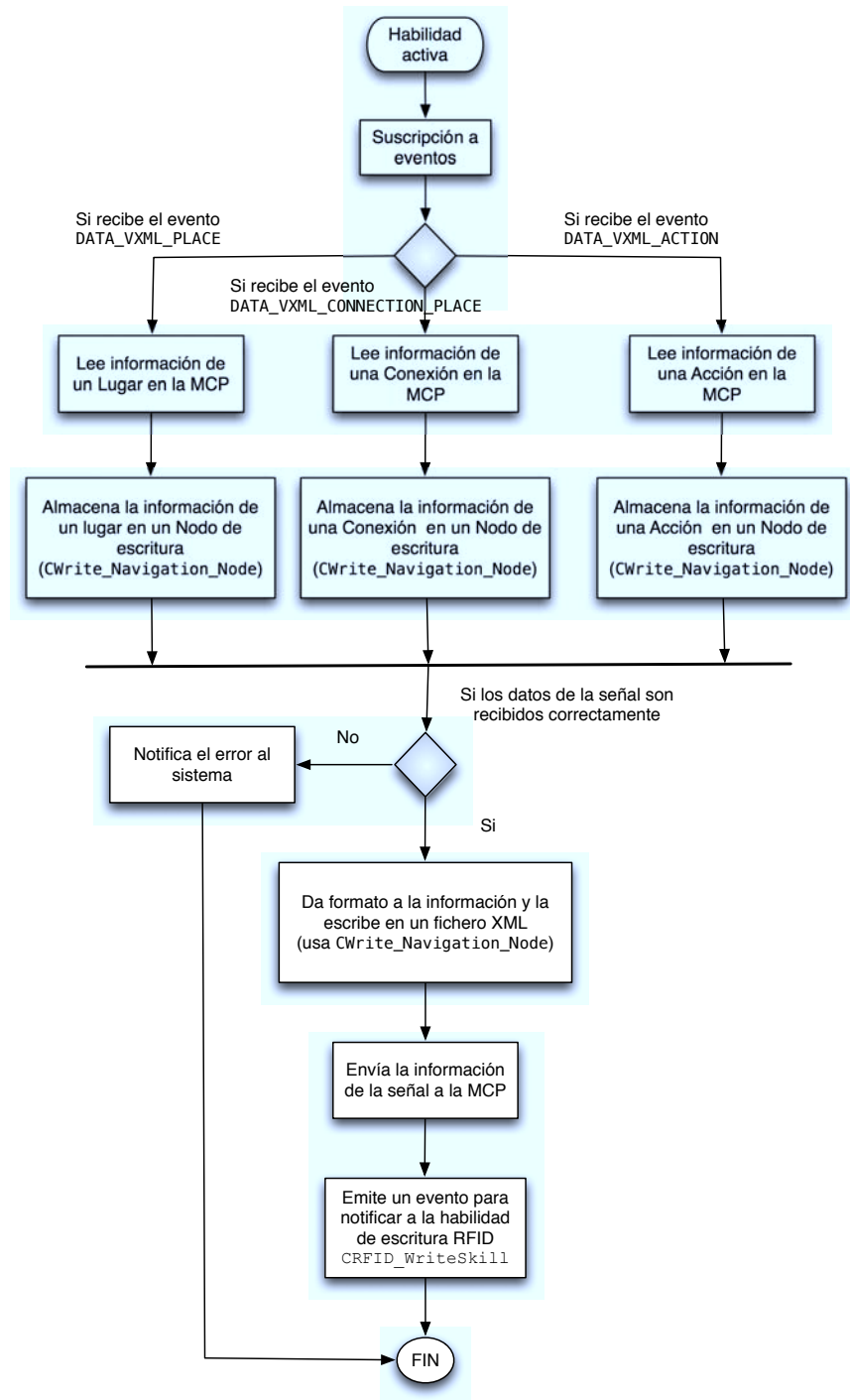
Para almacenar la información se ha creado un objeto llamado `CWrite_Navigation_Node` al que definiremos como *Nodo de Escritura*. Este objeto se encarga de recibir los datos almacenados por el Gestor de Diálogo en la Memoria a Corto Plazo y darles el formato adecuado para cada elemento de la señal (lugares y conexiones).

La clase `CWrite_Navigation_Node`, recibe los datos y verifica que se hayan recibido correctamente, si hay un error se notifica al sistema y se termina la ejecución de la habilidad.

Si ha recibido los datos correctamente, la habilidad da formato a la información y la escribe en un fichero XML (visto en el apartado 6.4). Posteriormente almacena la información en la Memoria a Corto Plazo y emite un evento para notificar a la habilidad de escritura de la señal (`CRFID_WriteSkill`) que los datos se encuentran disponibles para escribirlos físicamente en la señal, en este caso la etiqueta RFID (ver apartado 6.3.2).

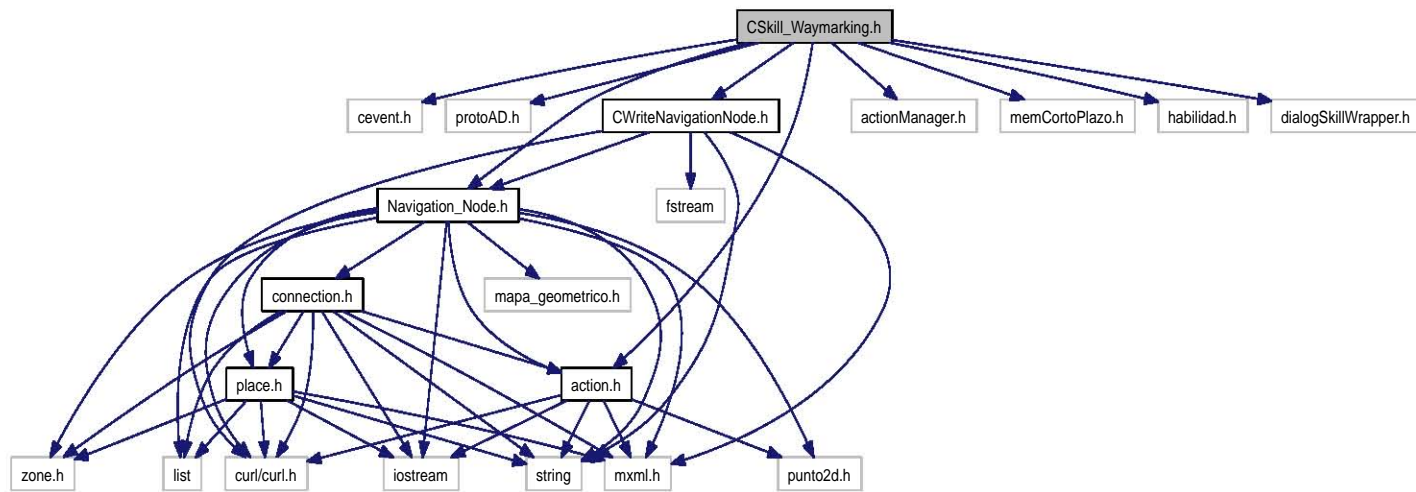
Las características más importantes de la Habilidad de Señalización son las siguientes:

1. Hereda de la clase abstracta *habilidad* [`habAD`], ya que debe formar parte de la arquitectura AD implementada en el robot Maggie.
2. Tiene un manejador de eventos [`CEventManager`] para recibir y enviar notificaciones desde y hacia otras habilidades del sistema.
3. Usa los objetos Nodo de Navegación y sus componentes: Lugar [`CPlace`], Zona [`CZone`] y Action [`CAction`].



**Fig. 6.23:** Diagrama de actividades de la Habilidad de Señalización CSkill\_Waymarking.

4. Usa una clase intermedia denominada CWrite\_Navigation\_Node, que escribe los datos de la señal en formato XML.



**Fig. 6.24:** Grafo de dependencias de la Habilidad de Señalización `CSkill_Waymarking`.

En la figura 6.24, se muestra el grafo de dependencia de clases de la habilidad `CSkill_Waymarking`. Como se puede observar depende de la clase base `habilidad` y del Gestor de Diálogos `dialogSkillWrapper`, este último lo usamos para facilitar la lectura de los datos en la Memoria a Corto Plazo.

Los datos que son leídos son a cadenas de caracteres con el siguiente formato:

- Para un **lugar**: *label \* ID \* option*.

Donde *label* corresponde al nombre del lugar, el *ID* su identificador y la *option* permite saber al sistema que un lugar tiene o no más zonas anidadas.

- Para una **conexión**: Como se ha explicado anteriormente, la información se recibe en dos partes:

1. El *lugar* que indica la conexión: *label<sub>c</sub> \* ID<sub>c</sub> \* option<sub>c</sub>*.
2. La *acción* que se debe realizar: *label<sub>a</sub> \* kind<sub>a</sub> \* code<sub>a</sub> \* option<sub>c</sub>*.

Para la acción los elementos son: nombre de la acción, tipo (topológica o geométrica), código (usado por el Manejador de Acciones) y la opción, que indica si hay o no más acciones que realizar para ir al lugar especificado previamente.

Cuando la habilidad `CSkill_Waymarking` recibe los datos, utiliza la clase `CWrite_Navigation_Node` para organizar los datos en el formato de una señal: Lugar y Conexiones.

Para un *lugar*, la clase almacenará los datos en una lista para así tener control de las *zonas* anidadas. Una vez que haya recibido todas las zonas anidadas que corresponden a un lugar lo asigna a un objeto `CPlace` (ver Algoritmos 6.1 y 6.2).

**Algoritmo 6.1:** Función *setPlaceData* de la clase *CWrite\_Navigation\_Node*

```
void CWrite_Navigation_Node::setPlaceData(string lb, string ident, int last){
    CZone zona_aux;
    zona_aux.setLabel(lb);
    zona_aux.setID(ident);
    this->list_place.push_back(zona_aux);
    /** Si last=1,
        indica que ya se deben guardar los datos de la lista como un lugar
        */
    if(last == 1){
        this->lugar_aux.setZones(list_place);
        this->nodo.setPlace(this->lugar_aux);
        this->list_place.clear();
    }
}
```

**Algoritmo 6.2:** Función *setPlace* del objeto *CNavigation\_Node*

```
void CNavigation_Node::setPlace(CPlace p){
    this->lugar = p;
}
```

Para una *conexión*, como se ha dicho anteriormente, se almacenará por separado sus dos componentes: el *lugar* y la *acción* o conjunto de *acciones*. La clase *CWrite\_Navigation\_Node*, almacenará los datos en una lista para así tener control de las *zonas* anidadas (al igual que *setPlace*). Una vez que haya recibido todas las zonas anidadas que corresponden a el lugar de la conexión, asigna la lista de zonas a un objeto *CConnection* (ver Algoritmo 6.3).

**Algoritmo 6.3:** Función *setPlaceConnectionData* de la clase *CWrite\_Navigation\_Node*

```
void CWrite_Navigation_Node::setPlaceConnectionData(string lb, string ident, int last){
    CZone zona_aux;
    zona_aux.setLabel(lb);
    zona_aux.setID(ident);
    this->list_place_connection.push_back(zona_aux);
    if(last == 1){
        this->connection_aux.setZones(list_place_connection);
        this->list_place_connection.clear();
    }
}
```

Para almacenar los datos de una *acción*, se han desarrollado dos funciones llamadas *setActionData* que dependiendo del tipo de acción, reciben diferentes parámetros.

- Si la acción es topológica: Recibe el nombre de la acción, el tipo, un código de identificación y un indicador de control para manejar la lista de acciones.
- Si la acción es geométrica: Además de los parámetros recibidos en el caso de ser una acción topológica, recibe una cadena que indica un fichero XML con los datos del mapa geométrico y un objeto *CPunto2d*, que indica la localización geométrica de la señal en el mapa indicado.

Posteriormente almacena la información de la acción o lista de acciones en el objeto *CConnection* (ver Algoritmos 6.4 y 6.5).

**Algoritmo 6.4:** Función *setActionData* de la clase *CWrite\_Navigation\_Node*

```
void CWrite_Navigation_Node::setActionData(string nombre, unsigned int tipo,
    unsigned int codigoID, int last){
    std::cout<<"asignando el nombre de la accion "<<nombre<<std::endl;
    this->accion_aux.asignarNombre(nombre);
    this->accion_aux.asignarTipo(tipo);
    this->accion_aux.asignarIdentificador(codigoID);
    this->list_action.push_back(this->accion_aux);
}
```



```

        if(last==1){
            this->connection_aux.setAction(this->list_action);
            this->list_action.clear();
        }
    }

void CWrite_Navigation_Node::setActionData(string nombre, unsigned int tipo,
                                           unsigned int codigoID, string mapa, CPunto2d localizacion, int last){
    this->accion_aux.asignarNombre(nombre);
    this->accion_aux.asignarTipo(tipo);
    this->accion_aux.asignarIdentificador(codigoID);
    this->accion_aux.setMapaGeometrico(mapa);
    this->accion_aux.asignarPunto(localizacion);
    this->list_action.push_back(this->accion_aux);
    if(last==1){
        this->connection_aux.setAction(this->list_action);
        this->list_action.clear();
    }
}

```

**Algoritmo 6.5:** Función *setConnectionData* de la clase *CWrite\_Navigation\_Node*

```

void CWrite_Navigation_Node::setConnectionData(int last){
    this->list_connections.push_back(this->connection_aux);
    if(last==1){
        this->nodo.setConnections(list_connections);
        this->list_connections.clear();
    }
}

```

Una vez adquiridos los datos y almacenados en las clases correspondientes, se escriben en un fichero XML a través de la función `CWrite_Navigation_Node::writeXML()`, con la que finalmente obtenemos los datos de la señal a escribir en formato XML, tal y como se muestra en 6.6.

**Algoritmo 6.6:** Código en XML de la señal

```

<?xml version="1.0"?>
<Node_Navigation>
    <NodeID></NodeID>
    <place>
        <zone>
            <label>Zone Label 1</label>
            <ID>ID.1</ID>
        </zone>
        <zone>
            <label>Zone Label 2</label>
            <ID>ID.1.2</ID>
        </zone>
        ...
    </place>
</Node_Navigation>

```

```

        <zone>
            <label>Zone Label n</label>
            <ID>ID1.2...n</ID>
        </zone>
    <connection>
        <place>
            <zone>
                <label>Zone Connection 1</label>
                <ID>ID_ZC1</ID>
            </zone>
        </place>
        <action>
            <nombre>Acción a realizar</nombre>
            <tipo>0</tipo>
            <codigo>2</codigo>
        </action>
    </connection>
</Node_Navigation>

```

Al almacenar los datos de la señal en el fichero XML, se almacena la dirección URL donde fue guardado en la Memoria a Corto Plazo y se emite el evento `NEW_RFID_DATA_TO_WRITE`, para que la habilidad suscrita a éste evento (en este caso `CRFID_WriteSkill`), lea los datos y ejecute el proceso de escritura física de la señal.

## 6.9. Discusión

A lo largo de este capítulo, se han presentado las diferentes etapas de programación e implementación de los elementos que componen el Sistema de Navegación para Robots Sociales Basado en Señales. La etapa inicial consistió en integrar en el robot Maggie, la plataforma donde se prueba este trabajo, los dispositivos sensoriales necesarios para realizar la lectura y escritura de las señales usadas en el sistema de navegación, es decir el sistema RFID. Seguidamente, basados en el diseño de la información de las señales, se realizó el estudio del almacenamiento lógico de los datos contenidos en las señales, de tal modo que sea fácil de procesar para su lectura y escritura. Posteriormente se implementaron los algoritmos diseñados para el sistema de navegación, se desarrolló el software de navegación en forma de habilidades, para que pudiese ser integrado dentro de la arquitectura de software usada en esta tesis, la arquitectura AD. Como el sistema de navegación contempla la interacción humano-robot, se diseñaron los diálogos de interacción considerando que la iniciativa la puede tomar el robot o una persona, con la finalidad de que el robot pueda adquirir fácilmente la información sobre las señales a crear o modificar. En el siguiente capítulo se detallarán las pruebas experimentales del sistema desarrollado e integrado en el robot social Maggie.

## 7. RESULTADOS EXPERIMENTALES

### 7.1. Introducción

En este capítulo se presenta una descripción de los diferentes experimentos que se han implementado para probar el funcionamiento del sistema de navegación planteado en esta tesis. La primera parte describirá los experimentos relacionados con la lectura de señales del entorno para la navegación, específicamente la implementación del algoritmo de navegación (Habilidad de navegación mediante señales). Posteriormente se describirán los experimentos del proceso de señalización a través de la interacción humano-robot. Finalmente se hará el análisis y discusión de los resultados obtenidos.

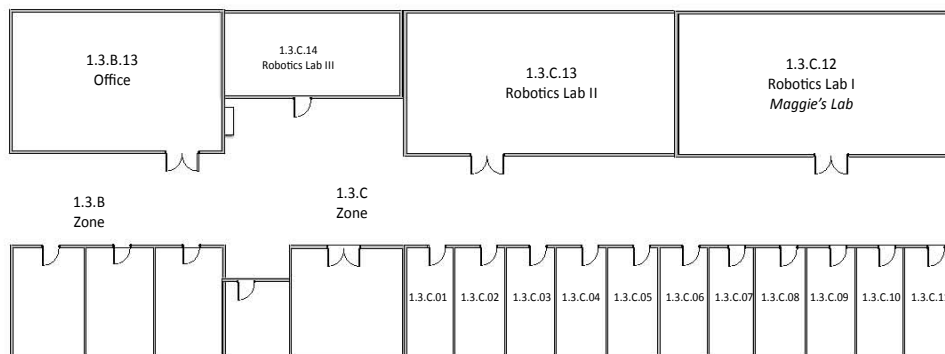
Como se ha dicho en el capítulo 6, el sistema de navegación se ha integrado en el robot Maggie, desarrollado en el RoboticsLab de la UC3M (Fig. 7.1).



**Fig. 7.1:** Robot Maggie, donde se ha integrado e implementado el sistema de navegación

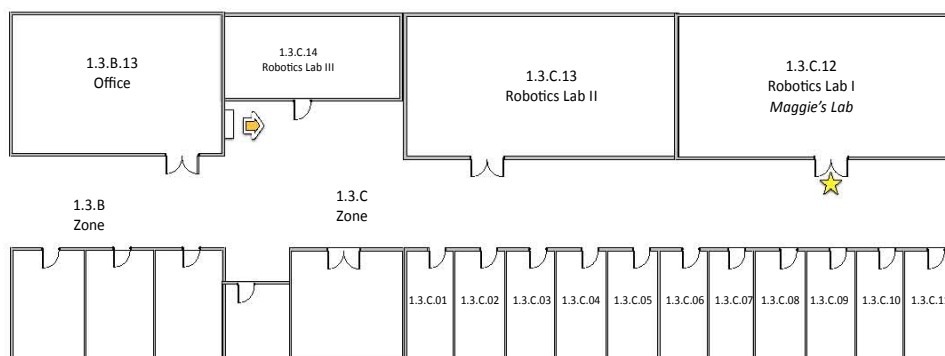
## 7.2. Habilidad de Navegación Mediante Señales

Los siguientes experimentos se realizaron en la Tercera Planta del Edificio Betancourt de la Universidad Carlos III de Madrid. En la figura 7.2 se muestra un plano del área de navegación del robot.



**Fig. 7.2:** Plano de parte de la Tercera Planta del Edificio Betancourt de la Universidad Carlos III de Madrid

### 7.2.1. Experimento 1



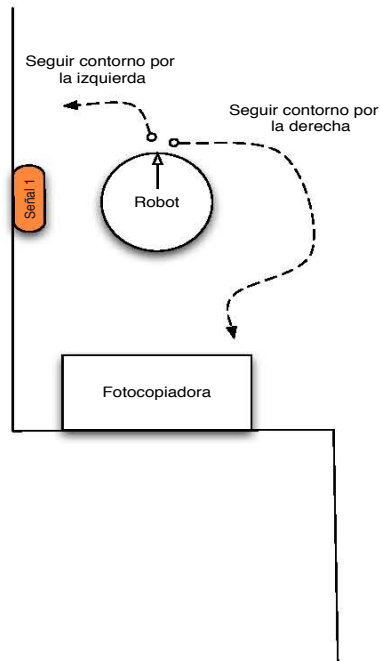
**Fig. 7.3:** Objetivo 1: El robot parte desde el punto indicado por la flecha y deberá llegar hasta la meta asignada, indicada con una estrella.

#### Objetivo

El robot debe ir a su laboratorio.

#### Descripción

Inicialmente, el robot se encuentra al lado de la fotocopiadora, que está ubicada en el pasillo de la zona C, del tercer piso del Edificio Betancourt (Figura 7.3). El

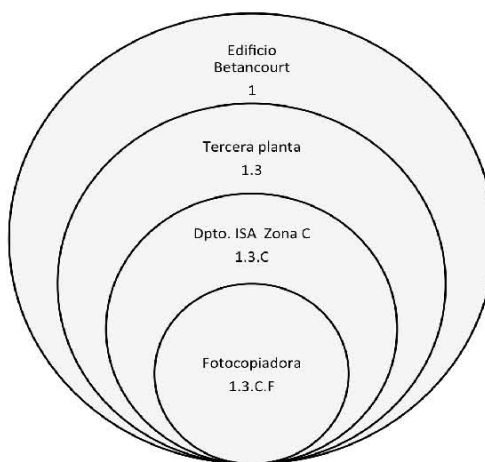


**Fig. 7.4:** Posición inicial del robot y posibles acciones que indica la señal al robot.

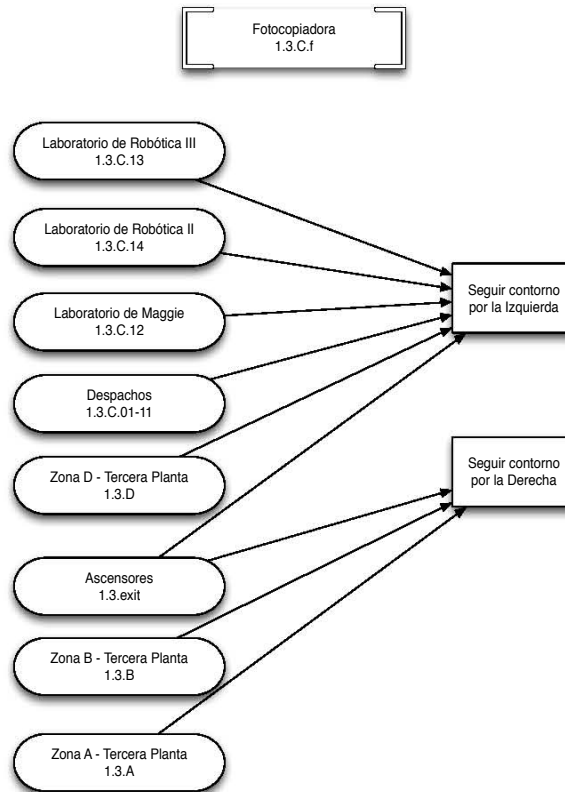
robot no sabe su ubicación hasta que lee la señal (etiqueta RFID) con la información (Figura 7.4). El robot debe moverse a través del pasillo y alcanzar la meta asignada.

#### Datos de entrada:

- Meta asignada: *Laboratorio de Maggie – 1.3.C.12*
- Primera señal detectada : Lugar: *Fotocopiadora – 1.3.C.f*



**Fig. 7.5:** Lugar que indica la señal inicial.



**Fig. 7.6:** Lugares hacia donde puede ir el robot y las acciones que debe realizar para ese fin.

**Proceso:** Inicialmente el robot detecta la primera señal, donde lee la información de su ubicación actual y los posibles lugares a donde puede ir desde ese sitio. La información que lee el robot está representada en las figuras 7.5 y 7.6. En la figura 7.5, se muestra un esquema de las zonas anidadas que componen el *lugar* que indica la primera señal. Como se ha explicado en capítulos anteriores, la zona más interna corresponde al sitio específico donde se encuentra inicialmente el robot.

La figura 7.6, muestra un esquema de las *conexiones* o diferentes *lugares* hacia donde puede ir el robot desde el sitio donde encuentra la señal. Los elementos de la izquierda corresponden a los *lugares* y a la derecha se encuentran las *acciones* que debería realizar para llegar a los lugares señalados.

Una vez que el robot haya leído la señal, éste sigue el algoritmo de navegación explicado en la sección 4.3. El robot compara la meta asignada con el lugar que indica la señal, si no coincide, procede a hacer la comparación con los lugares que indican las conexiones. Para el experimento, se puede observar que la meta asignada (*Laboratorio de Maggie – 1.3.C.12*) corresponde con una de las conexiones señaladas. El robot entonces lee la *acción* y activa la habilidad correspondiente “*seguir contorno por la izquierda*”. El sistema activa la habilidad y es ejecutada hasta que el robot encuentra una nueva señal, y posteriormente, continúa con el algoritmo de navegación.

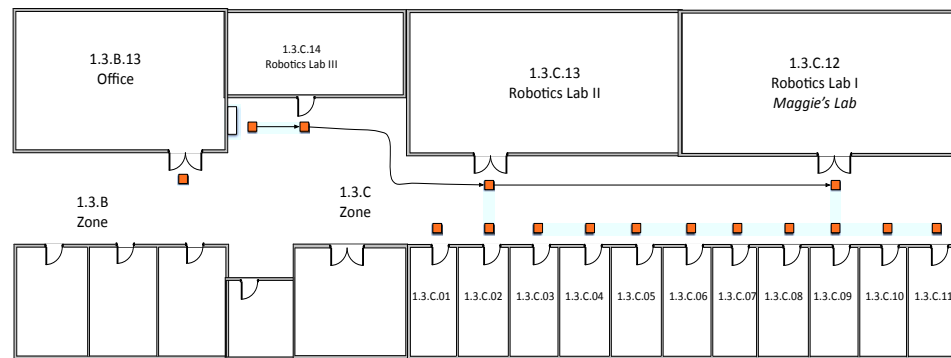
Este proceso se repite a medida que el robot navega y va encontrando señales

por el camino. La figura 7.8, representa el contenido de estas señales. De izquierda a derecha se muestra la información de cada una de las señales que el robot va leyendo a medida que va navegando. Para cada señal compara el lugar y/o las conexiones. Si la meta asignada corresponde con una conexión, el robot ejecuta la acción indicada. Cuando la meta asignada corresponde con el lugar, el robot se detiene, y se considera que ha completado la tarea de navegación.

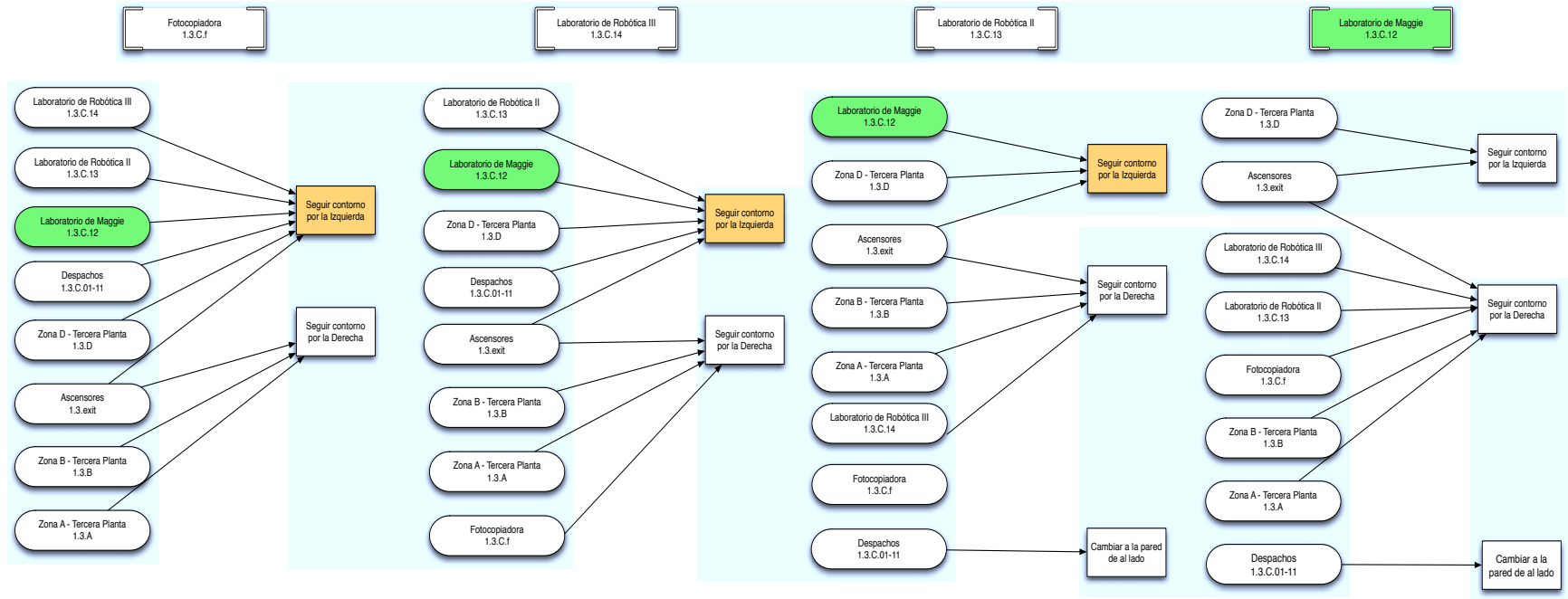
## Resultados

El robot se mueve por el entorno de acuerdo con las instrucciones señaladas en cada etiqueta RFID. Todas las etiquetas han sido detectadas por el robot y éste ha leído e interpretado correctamente la información, ejecutando las acciones necesarias para alcanzar la meta. Finalmente, el robot cumple satisfactoriamente la tarea de navegación asignada: *“Ir a su laboratorio”*.

La figura 7.7, muestra el mapa con la ruta del robot. Los recuadros naranja indican los puntos (*waypoints*), donde el robot detectó las señales de navegación (etiquetas RFID).



**Fig. 7.7:** Experimento 1: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot



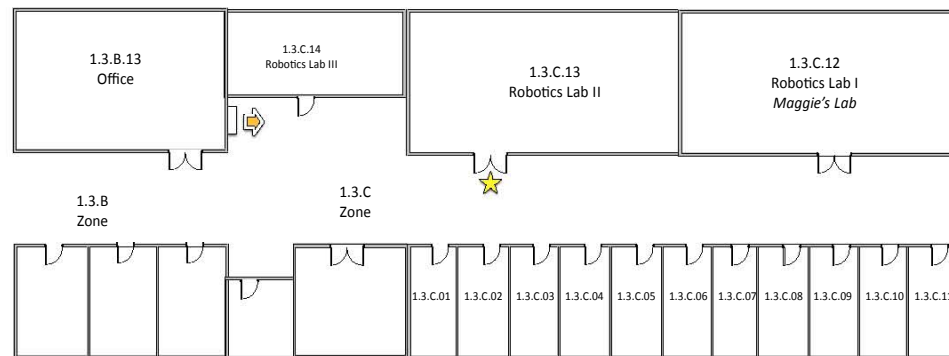
**Fig. 7.8:** Experimento 1: Información que lee el robot de las señales en la tarea de navegación. En verde se indica el lugar a donde debe ir el robot y en naranja la acción que ejecuta.



## 7.2.2. Experimento 2

### Objetivo

El robot debe ir al Laboratorio de Robótica II.



**Fig. 7.9:** *Objetivo 2: El robot parte desde el punto indicado por la flecha y deberá llegar hasta la meta asignada, indicada con una estrella.*

### Descripción

Al igual que en el experimento anterior, inicialmente, el robot se encuentra al lado de la fotocopiadora, que está ubicada en el pasillo de la zona C, del tercer piso del Edificio Betancourt (Figura 7.9). El robot no sabe su ubicación hasta que lee la señal (etiqueta RFID) con la información (Figura 7.4).

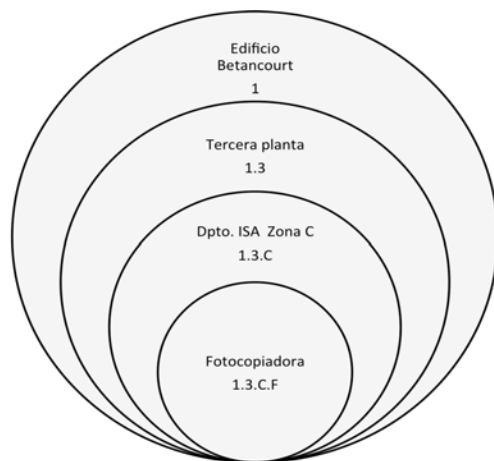
En este experimento se pretende probar el comportamiento del robot al encontrar un obstáculo móvil en el lugar destino, el cual impide la lectura de la señal.

### Datos de entrada:

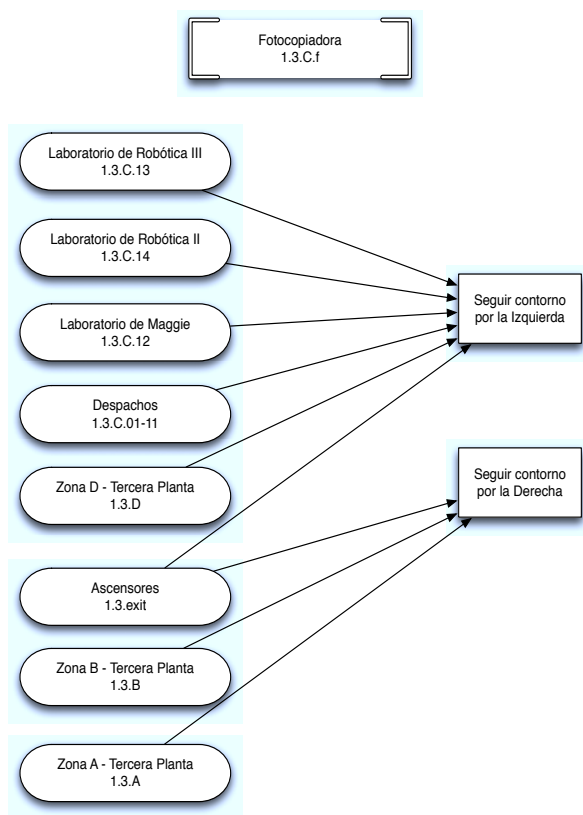
- Meta asignada: *Laboratorio de Robótica II – 1.3.C.13*
- Primera señal detectada : Lugar: *Fotocopiadora – 1.3.C.f*

**Proceso:** Inicialmente el robot detecta la primera señal, donde lee la información de su ubicación actual y los posibles lugares a donde puede ir desde ese sitio. La información que lee el robot está representada en las figuras 7.10 y 7.11.

El robot compara la meta asignada con el lugar que indica la señal, si no coincide, procede a hacer la comparación con los lugares que indican las conexiones. Para el experimento, se puede observar que la meta asignada (*Laboratorio de Robótica II – 1.3.C.13*) corresponde con una de las conexiones señaladas. El robot entonces lee la acción y activa la habilidad correspondiente “*seguir contorno por la izquierda*”. El sistema activa la habilidad y es ejecutada hasta que el robot encuentra una nueva señal, y posteriormente, continúa con el algoritmo de navegación. Este proceso se repite a medida que el robot navega y va encontrando señales por el camino.



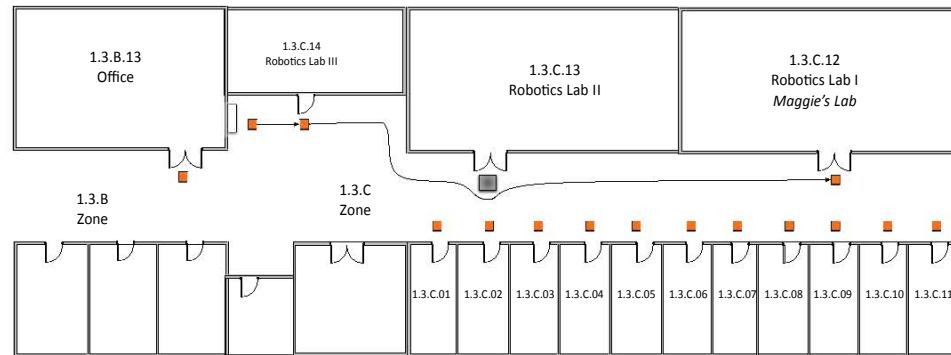
**Fig. 7.10:** Lugar que indica la señal inicial.



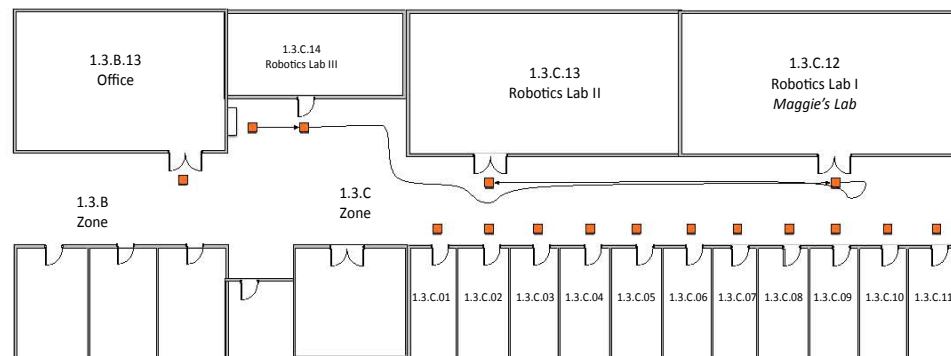
**Fig. 7.11:** Lugares hacia donde puede ir el robot y las acciones que debe realizar para ese fin.

A diferencia del Experimento 1, el robot se encuentra con un obstáculo al llegar a la meta y no logra leer la señal que le indica que se encuentra en el *lugar destino*, por lo que el robot continúa su trayecto ejecutando la acción que le había indicado la señal previa (Figura 7.12). Sin embargo, al leer la siguiente señal (ubicada en el 1.3.C.12),

encuentra su *lugar destino* en una de las conexiones, entonces el robot ejecuta la acción indicada “*sigue el contorno por la derecha*”. El robot se da la vuelta y al llegar a la siguiente señal, el obstáculo que había encontrado ya no se encuentra y entonces el robot puede leer la señal correctamente y llega a su destino satisfactoriamente (Figura 7.13).

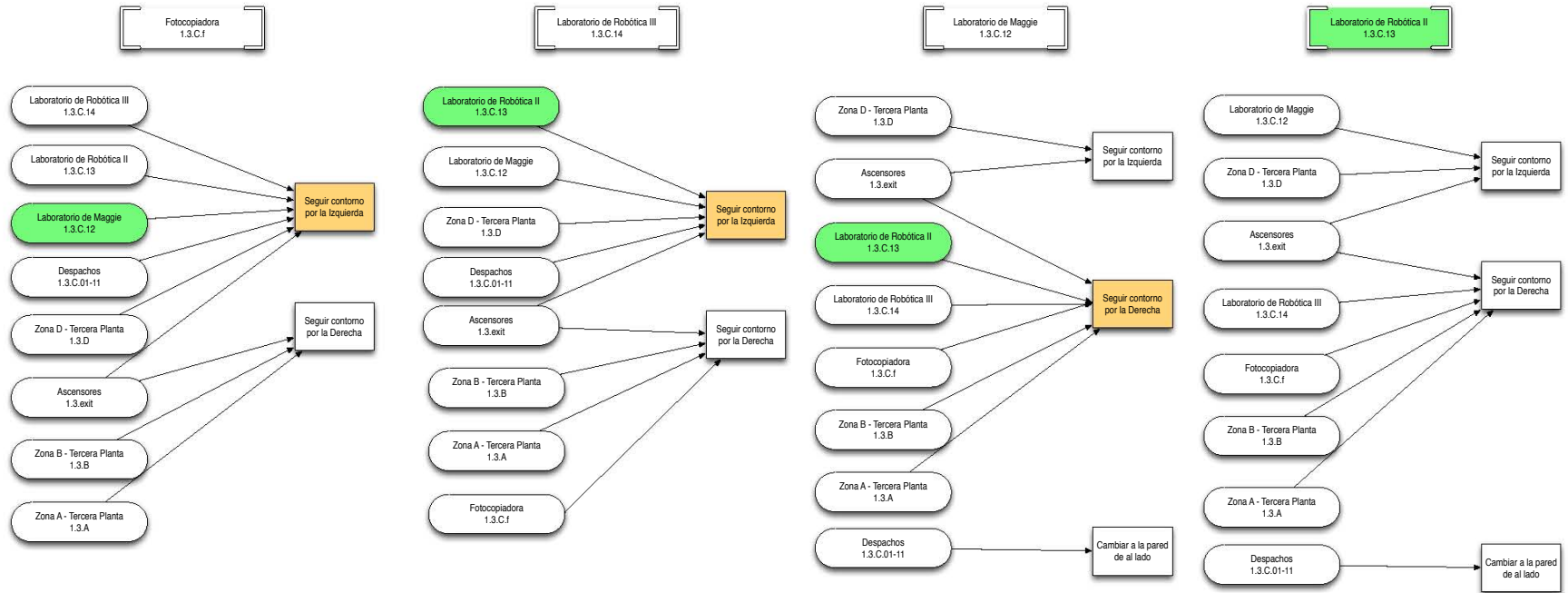


**Fig. 7.12:** Experimento 2: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot



**Fig. 7.13:** Experimento 2: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot

En la figura 7.14, se muestra un esquema del contenido de las señales de navegación que el robot encuentra en su camino a medida que va navegando. Los recuadros verdes indican la información que el robot usa para llegar a su destino.



**Fig. 7.14:** Experimento 2: Información que lee el robot de las señales en la tarea de navegación. En verde se indica el lugar a donde debe ir el robot y en naranja la acción que ejecuta.

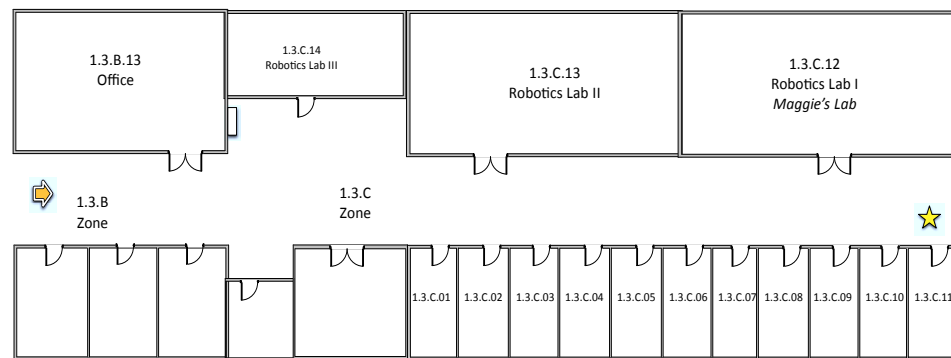
## Resultados

El robot se mueve por el entorno de acuerdo con las instrucciones señaladas en cada etiqueta RFID. A pesar de que inicialmente el robot encontró un obstáculo y no detectó la señal que indicaba que había llegado a la meta, finalmente, siguiendo las instrucciones de la siguiente señal, el robot alcanzó satisfactoriamente su destino y completó satisfactoriamente la tarea de navegación asignada: *“Ir al Laboratorio de Robótica II”*.

### 7.2.3. Experimento 3

#### Objetivo

El robot debe ir al Despacho 1.3.C.11.



**Fig. 7.15:** Objetivo 3: El robot parte desde el punto indicado por la flecha y deberá llegar hasta la meta asignada, indicada con una estrella.

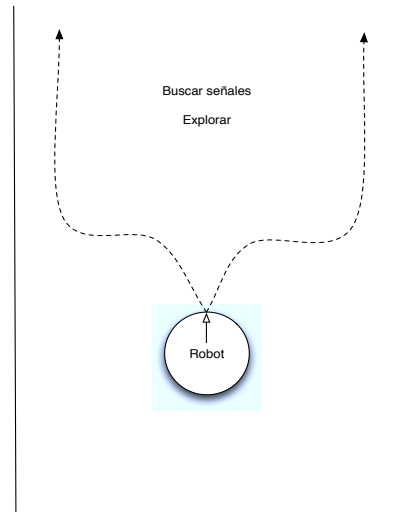
#### Descripción

Inicialmente, el robot se encuentra en el pasillo de la zona B, del tercer piso del Edificio Betancourt (Figura 7.15). El robot no sabe su ubicación, así que su primer objetivo es localizarse para posteriormente cumplir su objetivo principal *“Ir al Despacho 1.3.C.11”*.

#### Datos de entrada:

- Meta asignada: *Despacho – 1.3.C.11*
- El robot no sabe dónde se encuentra.

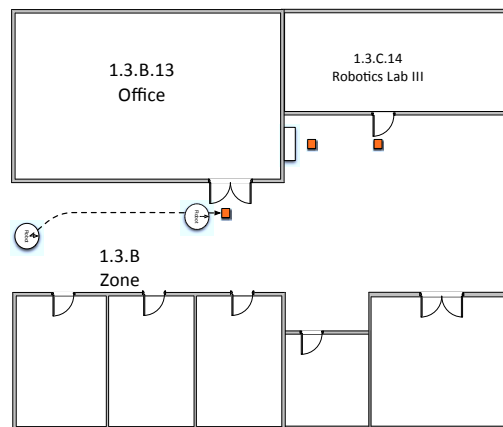
**Proceso:** Para lograr el objetivo de navegación, el robot necesita explorar el entorno, en búsqueda de una señal que le indique su ubicación. Para este experimento, el robot puede activar la habilidad de seguimiento de contorno (por la izquierda o por



**Fig. 7.16:** Posición inicial del robot y posibles movimientos que puede realizar para buscar señales en el entorno.

la derecha) ya que como se ha dicho anteriormente, las señales estarán ubicadas en la pared (Figura 7.16).

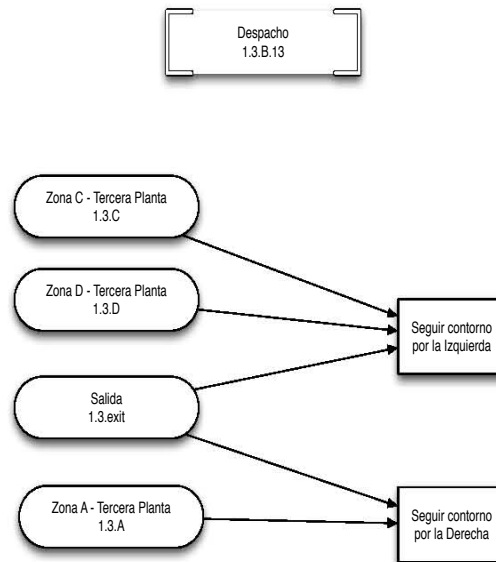
El robot, por defecto, comienza a explorar el entorno siguiendo el contorno por la izquierda. Cuando detecta la primera señal, el robot lee la información y se localiza (Figura 7.17). La información que lee el robot en la primera señal, se muestra en la Figura 7.18.



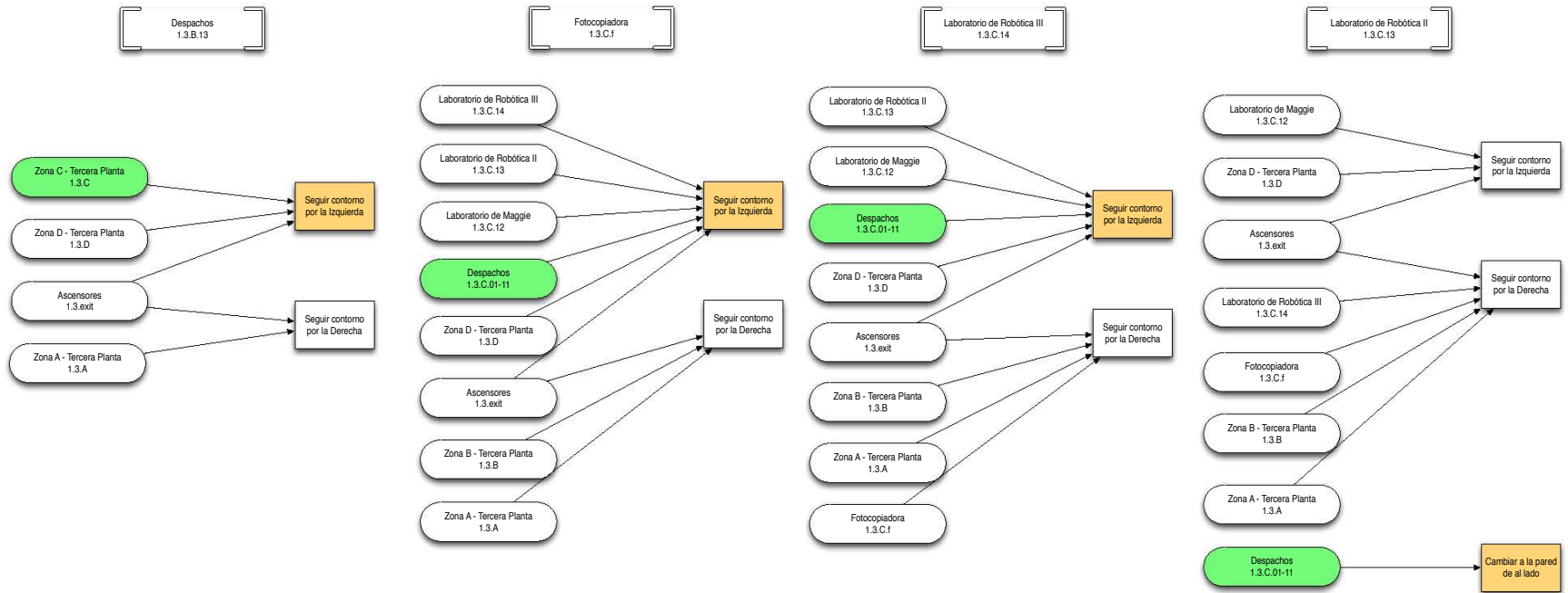
**Fig. 7.17:** El robot explora el entorno hasta encontrar una señal.

Una vez que el robot se ha localizado, comienza a ejecutar el algoritmo para la búsqueda de la meta asignada.

En las figuras 7.19 y 7.20, se muestra un esquema del contenido de las señales de navegación que el robot encuentra en su camino a medida que va navegando. Los recuadros verdes indican la información que el robot usa para llegar a su destino y los recuadros naranja la acción que realiza.

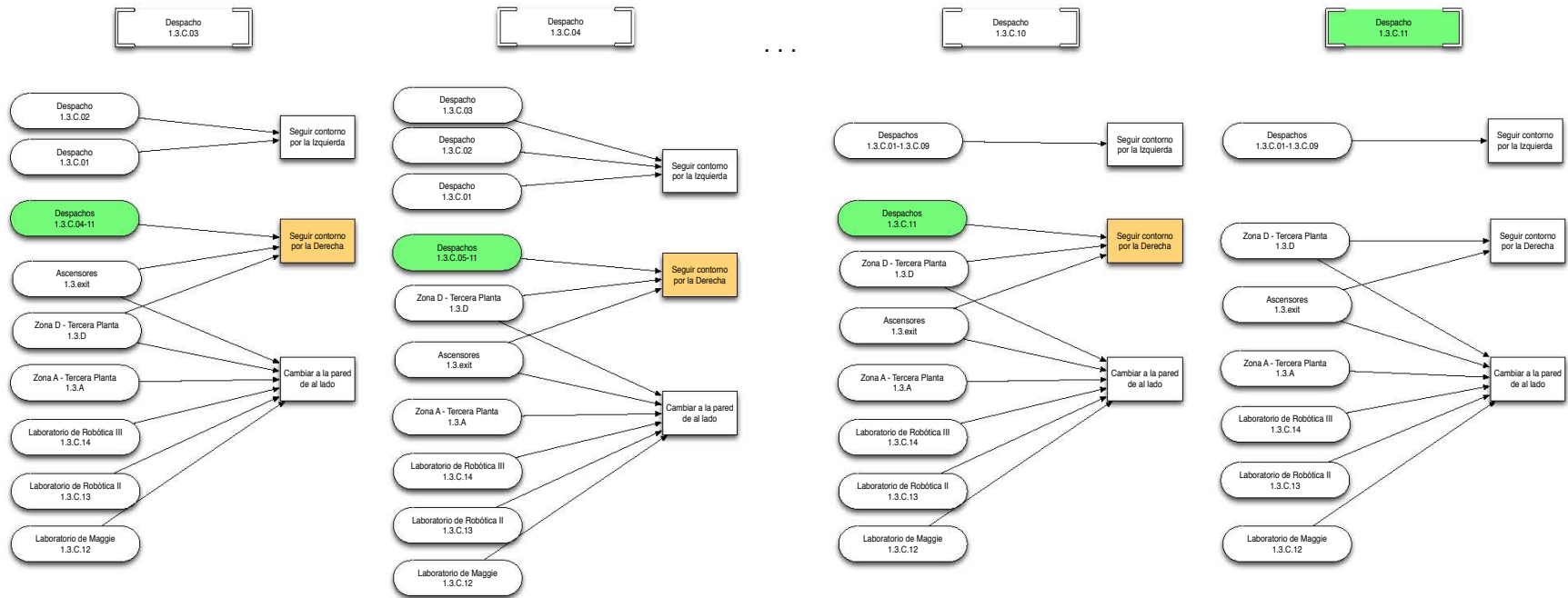


**Fig. 7.18:** Información que lee el robot al encontrar la primera señal.



**Fig. 7.19:** Experimento 3: Información que lee el robot de las primeras señales que encuentra en la tarea de navegación.

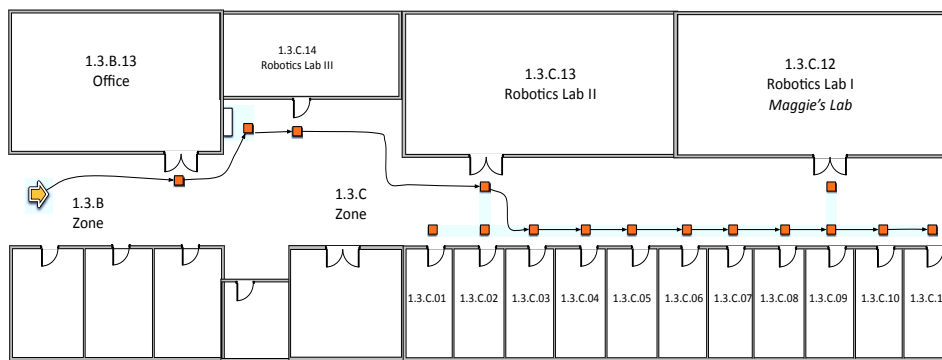




**Fig. 7.20:** Experimento 3: Resumen de la información de las señales 1.3.C.03 a la 1.3.C.11 con las acciones que realiza el robot.

## Resultados

En este experimento, a diferencia de los anteriores, la condición inicial establecía que el robot no sabía donde se encontraba inicialmente y no tenía a su alcance inmediato una señal RFID que le indicase su ubicación inicial, por lo que el robot entró en fase de *exploración*. En esta fase, el robot activó la habilidad “*Seguir contorno por la izquierda*” y encontró una señal que le localizó en el entorno. A partir de ese momento el robot procedió a ejecutar el Algoritmo de Navegación Mediante Señales. El robot se movió por el entorno ejecutando las acciones indicadas por las señales. Finalmente el robot cumplió satisfactoriamente la tarea de navegación “*Ir al Despacho 1.3.C.11*”. La figura 7.21, muestra el recorrido realizado por el robot para llegar a la meta.



**Fig. 7.21:** Experimento 3: Recorrido realizado por el robot.

### 7.2.4. Experimento 4

#### Objetivo

El objetivo es el mismo que en el Experimento 3. El robot debe ir al Despacho 1.3.C.11, pero encuentra un obstáculo al leer una señal en el camino.

#### Descripción

Inicialmente, el robot se encuentra en el pasillo de la zona B, del tercer piso del Edificio Betancourt (Figura 7.15). El robot no sabe su ubicación, así que, al igual que en el experimento anterior, su primer objetivo es localizarse para posteriormente cumplir su objetivo principal “*Ir al Despacho 1.3.C.11*”.

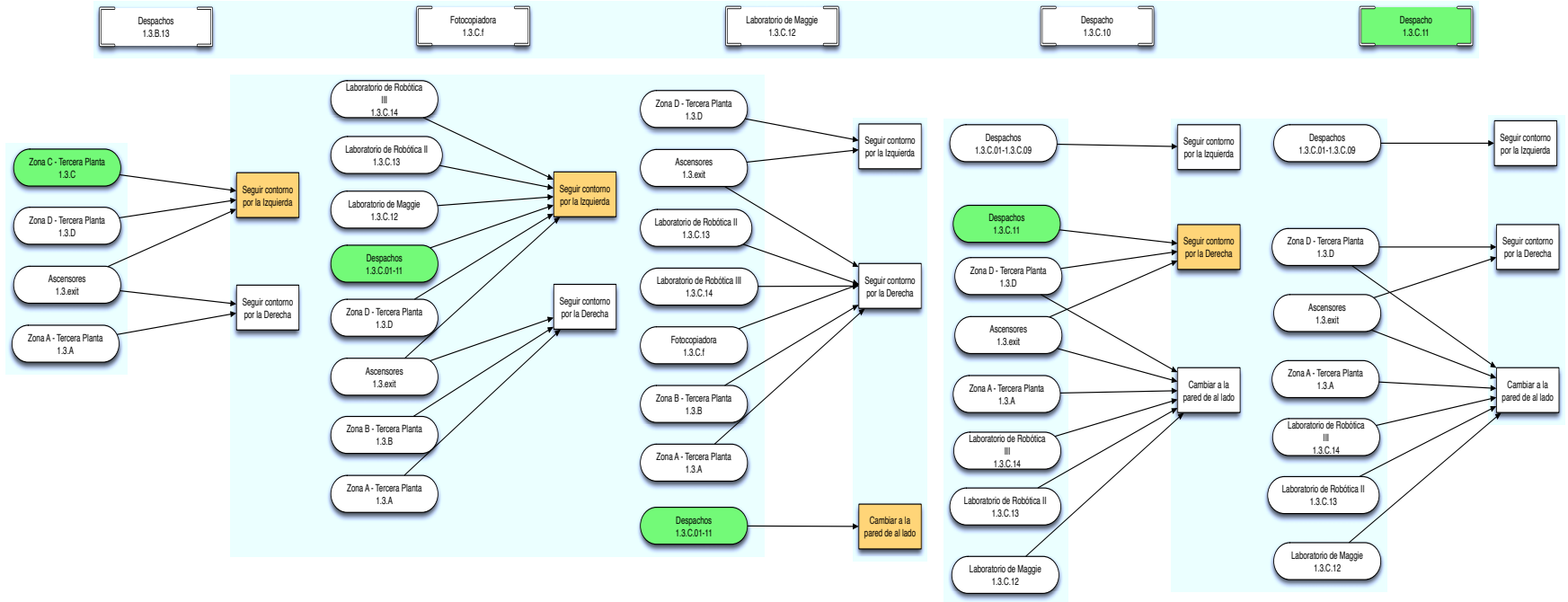
#### Datos de entrada:

- Meta asignada: *Despacho – 1.3.C.11*
- El robot no sabe dónde se encuentra.

**Proceso:** Para lograr el objetivo de navegación, el robot necesita explorar el entorno, en búsqueda de una señal que le indique su ubicación. El robot realizó el proceso tal y como se explicó en el Experimento 3.

A diferencia del experimento anterior, el robot no puede leer la señal ubicada en el Laboratorio 1.3.C.13, entonces éste continúa ejecutando la acción previa: *“Seguir contorno por la izquierda”*, que ha leído en la señal anterior (1.3.C.14) . Al encontrar la siguiente señal (1.3.C.12) el robot busca en las conexiones una que corresponda con el destino asignado y ejecuta la acción correspondiente.

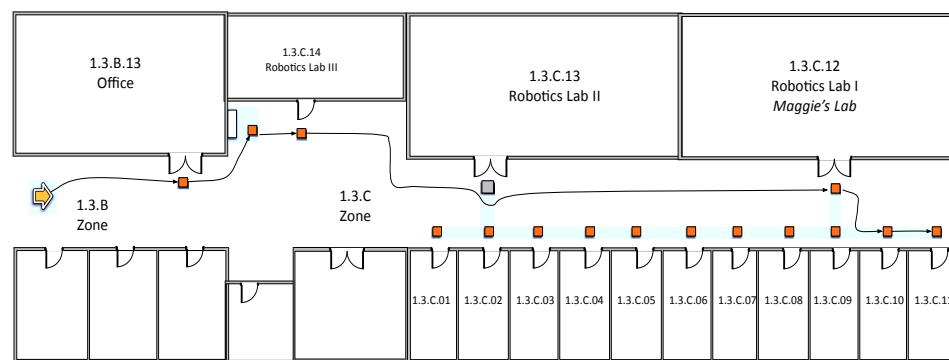
En la figura 7.22, se muestra el contenido de las señales que el robot encuentra a lo largo de su recorrido y las acciones que sigue para llegar a su destino.



**Fig. 7.22:** Experimento 4: Información de las señales que encuentra el robot.

## Resultados

En este experimento, al igual que el anterior, la condición inicial establecía que el robot no sabía donde se encontraba inicialmente y no tenía a su alcance inmediato una señal RFID que le indicase su ubicación inicial, por lo que el robot entró en fase de *exploración*. En esta fase, el robot activó la habilidad “*Seguir contorno por la izquierda*” y encontró una señal que le localizó en el entorno. A partir de ese momento el robot procedió a ejecutar el Algoritmo de Navegación Mediante Señales. El robot se movió por el entorno ejecutando las acciones indicadas por las señales, pero encontró un obstáculo que imposibilitó la lectura de la señal ubicada en el Laboratorio de Robótica II (1.3.C.13). El robot continuó ejecutando la acción indicada en la señal previa (ubicada en 1.3.C.14) y nuevamente encontró una señal que le indicó como llegar a su destino. Finalmente el robot cumplió satisfactoriamente la tarea de navegación “*Ir al Despacho 1.3.C.11*”. La figura 7.23, muestra el recorrido realizado por el robot. Se puede observar gráficamente su comportamiento al encontrar el obstáculo y el modo en que llega a la meta asignada.



**Fig. 7.23:** Experimento 4: Recorrido realizado por el robot y su comportamiento al encontrar un obstáculo que impide la lectura de una señal.

### 7.2.5. Experimento 5

#### Objetivo

El robot debe ir a su laboratorio (1.3.C.12) y apagar la televisión.

#### Descripción

Inicialmente, el robot se encuentra al lado de la fotocopidora, que está ubicada en el pasillo de la zona C, del tercer piso del Edificio Betancourt (Ver Figura 7.3). El robot no sabe su ubicación hasta que lee la señal (etiqueta RFID) con la información. El robot debe moverse a través del pasillo, entrar al laboratorio 1.3.C.12 y apagar la TV.

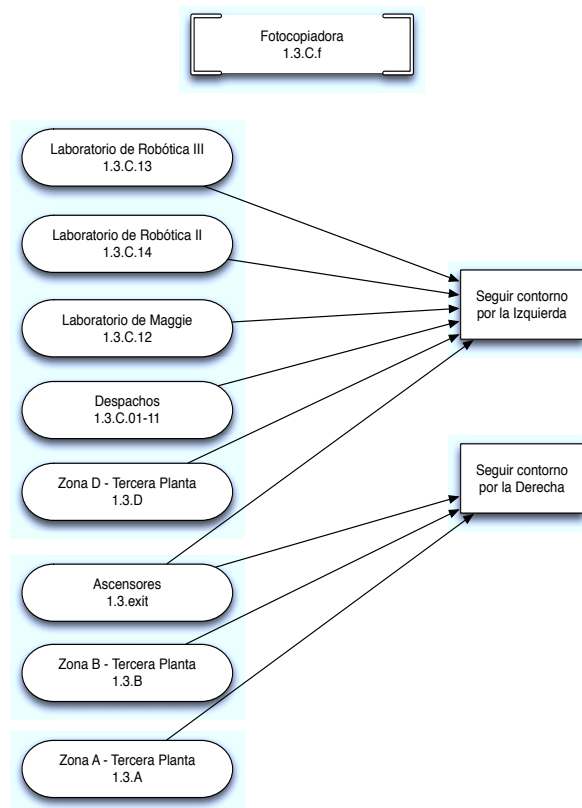
**Datos de entrada:**

- Meta asignada: *Televisión del Laboratorio de Maggie – 1.3.C.12.TV*
- Primera señal detectada : Lugar: *Fotocopiadora – 1.3.C.f*

**Proceso:** En este experimento la tarea del robot se compone de 2 objetivos:

1. “Ir a la Televisión del Laboratorio de Maggie”.
2. “Apagar la Televisión”.

**Objetivo 1: “Ir a la Televisión del Laboratorio de Maggie”:** Como se ha dicho previamente, el robot inicialmente se encuentra al lado de la fotocopiadora (1.3.C.f). El robot lee la primera señal y compara las conexiones con la meta asignada. Recordemos la información almacenada en la primera señal que lee el robot (Figura 7.24).



**Fig. 7.24:** Lugares hacia donde puede ir el robot y las acciones que debe realizar para ese fin.

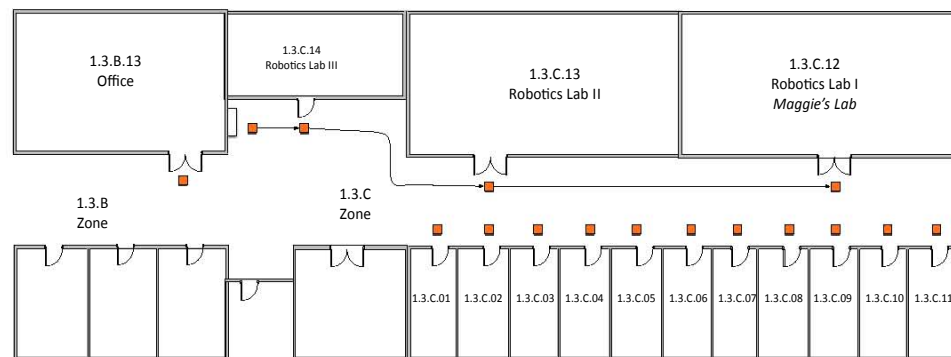
Como se puede observar, no hay una coincidencia exacta con el identificador del punto destino “1.3.C.12.TV”; por lo que el sistema busca entonces una zona genérica que lo aproxime a la meta, en este caso, la conexión más indicada es la del “Laboratorio de Maggie 1.3.C.12” (Figura 7.25).

# 1.3.C.12.TV

## 1.3.C.12

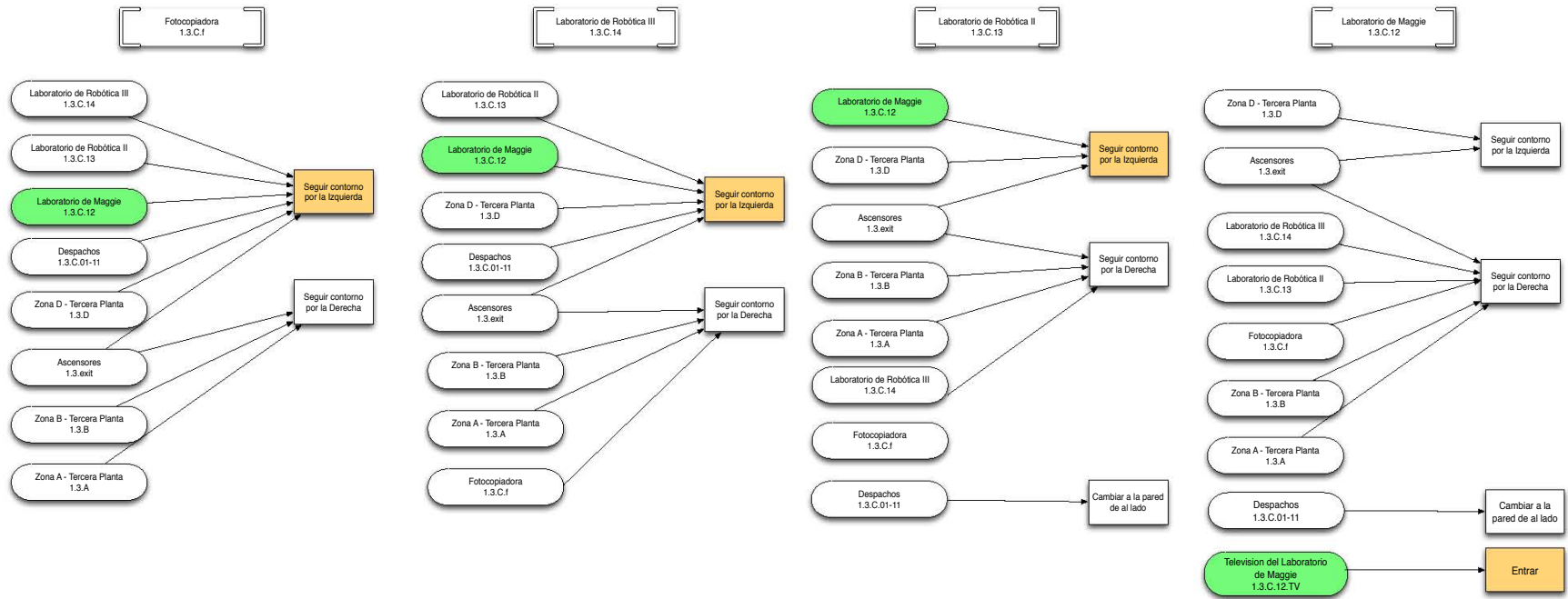
**Fig. 7.25:** Identificador del punto destino y de la zona genérica que toma en cuenta el robot.

A partir de aquí, el sistema se comporta de igual modo que en el Experimento 1 (Sección 7.2.1). El robot navega por el entorno guiándose por las señales RFID que encuentra. El robot se detiene cuando encuentra su objetivo parcial: *Laboratorio de Maggie – 1.3.C.12*. (Figura 7.26).



**Fig. 7.26:** Experimento 5: Plano del entorno con las señales (etiquetas RFID) y la ruta realizada por el robot hasta llegar al Laboratorio 1.3.C.12

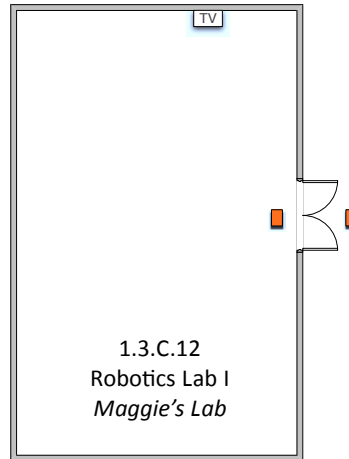
En la figura 7.27, se puede observar un resumen de la información que lee el robot en las señales que va encontrando durante su recorrido, y las indicaciones que va siguiendo hasta llegar al Laboratorio 1.3.C.12. Una vez que llega al Laboratorio 1.3.C.12, el robot busca una conexión que corresponda a su punto de destino, como se puede ver en la figura, existe una conexión que le indica al robot “Entrar” al laboratorio.



**Fig. 7.27:** Experimento 5: Información que lee el robot de las señales en la tarea de navegación. En verde se indica el lugar a donde debe ir el robot y en naranja la acción que ejecuta.

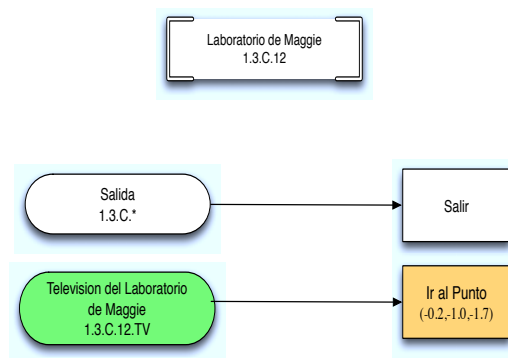


Para este experimento, se colocó una señal dentro del Laboratorio 1.3.C.12, justo en la entrada (7.28). En la señal se agregó una conexión adicional, indicando la posición dentro del laboratorio donde el robot puede situarse frente a la televisión.



**Fig. 7.28:** Experimento 5: Ubicación de las señales en el Laboratorio 1.3.C.12

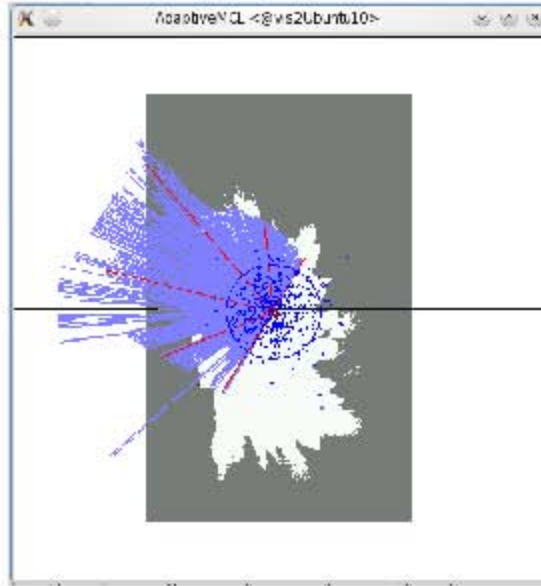
La figura 7.29, muestra el esquema de la información almacenada en la señal del interior del laboratorio. La conexión que se ha añadido especifica el lugar hacia donde debe ir el robot (*Televisión del Laboratorio de Maggie*) y la acción que debe realizar el robot para llegar a ese sitio (*Ir al Punto*  $(-0.2, -1.0, -1.7)$ ).



**Fig. 7.29:** Experimento 5: Información de la señal situada en el Laboratorio 1.3.C.12

A diferencia de los experimentos anteriores, en este experimento se está usando una *acción geométrica*. Como se comentó en la sección 3.3.1, la información geométrica es necesaria cuando la información topológica es insuficiente o si nos encontramos en entornos no estructurados, en este caso concreto, el robot necesita entrar a su laboratorio (donde el entorno se caracteriza por tener un espacio rodeado de sillas y mesas de trabajo) e ir a un punto geométrico concreto, para que el control infrarrojos del robot tenga acceso a la Televisión y pueda cumplir con el siguiente objetivo “Apagar la Televisión”.

La figura 7.30 muestra el mapa geométrico del laboratorio 1.3.C.12, realizado con el algoritmo *AMCL* (*Adaptive Monte-Carlo Localization*) [Fox et al., 1999] implementado en el proyecto *Player* [Gerkey et al., 2003].



**Fig. 7.30:** Mapa geométrico del Laboratorio 1.3.C.12. El robot no está localizado geométricamente.

A continuación se muestra el código en XML de la conexión geométrica que indica al robot moverse a un punto geométrico específico del mapa.

**Algoritmo 7.1:** Información en XML de la conexión hacia la TV del Laboratorio de Maggie – 1.3.C.12.TV

```

1 <connection>
2   <place>
3     <zone>
4       <label>Televisión del Laboratorio de Maggie</label>
5       <ID>1.3.C.12.TV</ID>
6     </zone>
7   </place>
8   <action>
9     <nombre>
10    GTP (-0.2,-1.0,-1.7)
11    </nombre>
12    <tipo>
13    1
14    </tipo>
15    <codigo>
16    3
17    </codigo>
18    <GeometricMap>
19      <direccion>
20      file:///usr/local/AD.trunk/robot/longTermMemory/robot_maggie_lab.cfg

```

```

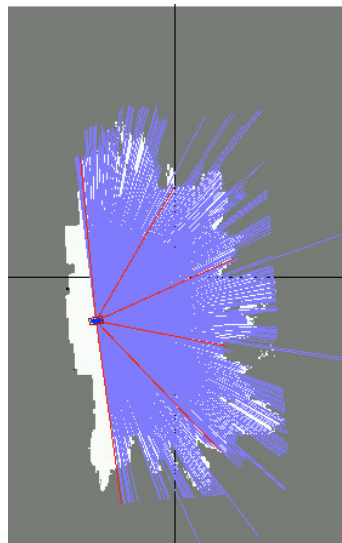
21         </direccion>
22         <localizacion>(-1.9,0.33,0.0)</localizacion>
23     </GeometricMap>
24 </action>
25 </connection>

```

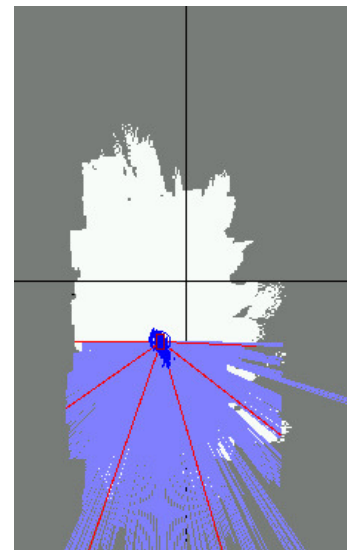
En el código se puede observar los diferentes elementos que componen una conexión geométrica:

- **<nombre>**: Especifica el nombre de la acción. En este caso **GTP**  $(-0.2, -1.0, -1.7)$ , indica que el robot activará la habilidad “Go to Point” (Ir a un punto), hacia la coordenada  $x = -0.2$ ,  $y = -1.0$  y  $\theta = -1.7$ , expresados en metros, metros y radianes respectivamente.
- **<GeometricMap>**: Para que el robot active la habilidad “Go to Point”, el sistema requiere el mapa geométrico del entorno para que el robot se localice. Este mapa se ha hecho previamente y se ha almacenado en la URL indicada en el campo **<direccion>**.
- **<localizacion>**: Este campo indica al robot la localización geométrica con respecto al mapa geométrico para el momento que detecta la señal. Esto hace que el robot se localice inmediatamente y pueda ejecutar la habilidad “Go to Point” sin tener que hacer una exploración previa del entorno.

En las figuras 7.31 y 7.32, se muestra la localización geométrica del robot al detectar la señal colocada en la entrada del laboratorio y cuando llega a la coordenada frente al televisor.



**Fig. 7.31:** Robot localizado geométricamente en la entrada del Laboratorio 1.3.C.12



**Fig. 7.32:** Robot localizado geométricamente frente al Televisor del Laboratorio 1.3.C.12

**Objetivo 2: “Apagar la Televisión”:** Una vez que el robot alcanza el primer objetivo “*Ir a la Televisión del Laboratorio de Maggie*”, el robot activa la Habilidad de Control Remoto a través de Infrarrojos [Salichs et al., 2009]. Esta habilidad habilita el receptor/transmisor de infrarrojos integrado en el robot Maggie, con la finalidad de manejar diferentes aparatos o dispositivos electrónicos que pueden estar colocados en el entorno, en este caso interesa manejar la Televisión.

Cuando se asigna la meta de la *Habilidad de Navegación mediante Señales*, el sistema se suscribe al evento `SUCCESSFUL_SKILL_WAYFINDING`, que notifica que la tarea de navegación se ha completado satisfactoriamente. Cuando el evento es recibido, el sistema ejecuta entonces el comando “*Apagar la Televisión*” almacenado en el momento que se asignó la tarea.

En el algoritmo 7.2 se muestra un resumen del código desarrollado. En la línea 3, se muestra la suscripción al evento que notifica que el robot ha llegado al punto geométrico especificado. Al suscribirse, se pasa como parámetro el comando que se desea enviar al infrarrojo (`cmd_ir`). Cuando el sistema recibe el evento, ejecuta la función *ejecutar\_acción* (línea 8), esta función almacena en la Memoria a Corto Plazo del sistema el comando para apagar la TV y emite el evento `MANDO_IR` para activar la Habilidad de Control Remoto (línea 26).

**Algoritmo 7.2:** *Función ejecutar\_accion para apagar la TV*

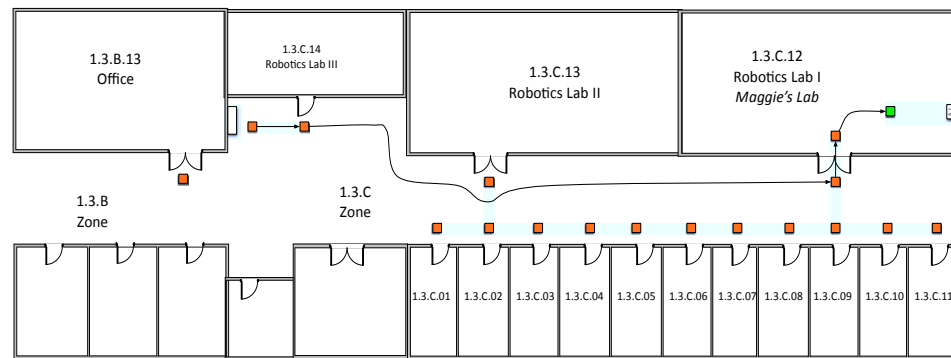
```

1  ...
2
3  emisor.subscribe(SUCCESSFUL_SKILL_WAYFINDING, ejecutar_accion,
4                    (void*)(cmd_ir.c_str()));
5
6  ...
7
8  void ejecutar_accion(void* aux,int n){
9      char *command;
10     command = (char*) aux;
11     CmemCortoPlazo memCP;
12     struct estructura_codigo codigo;
13     CItem item(sizeof(estructura_codigo));
14     item.asignarID(ID_COMANDO_IR_MCP);
15     item.asignarFecha(0);
16     if ( memCP.RegistrarDato(item)){
17         std::cerr << "Error al registrar el item\n";
18     }
19     strcpy(codigo.remoto, dispositivo);
20     strcpy(codigo.comando, command);
21     item.asignarDato((char *)&codigo, sizeof(estructura_codigo));
22     if (memCP.ColocarDato(item)) {
23         std::cerr << "Error colocando dato\n";
24         return ;
25     }
26     emisor.emit(MANDO_IR);
27     std::cout<<"Emitiendo MANDO_IR\n";
28     return;
29 }
```

## Resultados

En este experimento se le asignó al robot una meta de navegación y una tarea que debía realizar al llegar al punto destino. El robot se movió por el entorno de acuerdo con las instrucciones señaladas en cada etiqueta RFID. Todas las etiquetas han sido detectadas por el robot y éste ha leído e interpretado correctamente la información, ejecutando las acciones necesarias para alcanzar la meta. A diferencia de los experimentos anteriores, en la última fase de navegación el robot requería ir a un punto geométrico específico. El robot se localizó geométricamente en el mapa geométrico señalado por la etiqueta y llegó satisfactoriamente al punto geométrico deseado para situarse frente a la televisión y así logró ejecutar satisfactoriamente la siguiente sub-tarea “Apagar la Televisión”.

En la figura 7.33 se muestra el mapa con la ruta del robot. Los recuadros naranja indican los puntos (*waypoints*), donde el robot detectó las señales de navegación (etiquetas RFID).



**Fig. 7.33:** Experimento 5: Plano del entorno con las señales (etiquetas RFID) y el recorrido realizado por el robot.

### 7.3. Señalización del Entorno

En esta sección se presentan los diálogos diseñados para probar el sistema de señalización mediante interacción humano-robot. El objetivo de cada diálogo es que el robot pueda adquirir la información necesaria para señalar la etiqueta correspondiente. Los ejemplos mostrados a continuación, fueron realizados para señalar la Tercera Planta del Edificio Betancourt de la Universidad Carlos III de Madrid.

#### 7.3.1. Definición de la Gramática Semántica

Como se explicó previamente en la sección 5.3, la gramática permite al reconocedor devolver valores relacionados con el significado de las frases que reconoce. Para estos experimentos, se diseñó una gramática que describe la información sobre los lugares y conexiones para señalar la Tercera Planta del Edificio Betancourt de la UC3M.

```

1  #ABNF 1.0 ISO-8859-1;
2
3  language es-ES;
4  tag-format <loq-semantics/1.0>;
5
6  public $root = $waymarking;
7
8  public $waymarking = $adjetivos | $ids | $op | $acciones |
9                      $codigos | $tipodeaccion;
10
11 public $lugares = ("edificio betancourt" | "tercera planta" |
12 "departamento de ingeniería de sistemas y automática zona c" |
13 "laboratorio de maggie" | "laboratorio de robotica 2" |
14 "departamento de ingeniería de sistemas y automática zona b" |
15 "departamento de ingeniería eléctrica" |
16 "laboratorio de robótica 3" | "servicios" | "televisión" | "tele" |
17 "tercera planta" | "zona" | "despacho" ){<@lugares $value>};
18
19 public $ids = ("uno" | "1 punto 3" | "1 punto 3 c" | "1 punto 3 c 12"
20 | "1 punto 3 c 13" | "1 punto 3 b" | "1 punto 3 d" | "1 punto 3 c 14"
21 | "1 punto 3 c s" | "1 punto 3 c 12 tv" | ){<@ids $value>};
22
23 public $op = ("si":si | "no":no | "okey":si | "vale":si | "negativo":no
24 | "de acuerdo":si | "incorrecto":no ){<@op $value>};
25
26 public $acciones = ("sigue el contorno por la derecha"
27 | "sigue el contorno por la izquierda"
28 | "usa la habilidad ir a un punto" | "entrar"){<@acciones $value>};
29
30 public $adjetivos = [("esto es"|"estas en"|"aquí es"|"puedes ir a"|"hacia")|$lugares;
31
32 public $codigos = ("0" | "1" | "2" | "3"){<@codigos $value>};
33
34 public $tipodeaccion = ("si" | "no"){<@tipodeaccion $value>};

```

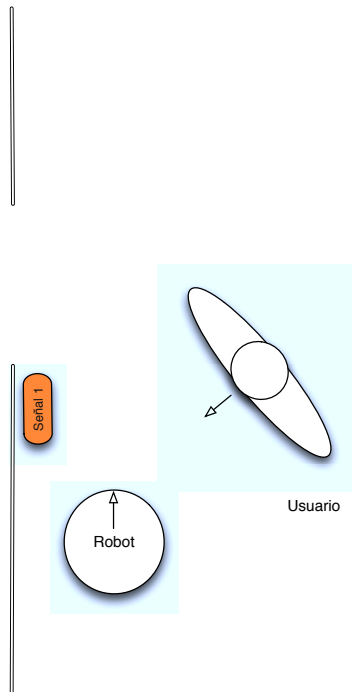
Como se puede observar, se ha diseñado la gramática semántica en base a las reglas gramaticales básicas explicadas en la sección 5.3.1.

- En **\$lugares**, se indica los posibles nombres de los lugares y/o zonas que se deben señalar. Estos son los sitios a los que puede ir el robot en el entorno donde se ha realizado el experimento.
- En **\$ids**, se encuentran los identificadores alfanuméricos de los lugares que se quieren señalar. Los identificadores deben corresponderse con las normas de señalización de edificios de la Universidad Carlos III de Madrid<sup>1</sup>.
- En **\$acciones**, se definen las frases que el usuario dirá al robot para indicarle los movimientos o acciones que debe ejecutar para ir de un lugar a otro.

Los diálogos realizados buscan responder las siguientes interrogantes, necesarias para conformar la información de una señal: ¿Dónde estoy?, ¿Hacia dónde puedo ir? y ¿Qué debo hacer para llegar a ese sitio?.

### 7.3.2. El robot toma la iniciativa

Este experimento es el más simple, el robot puede tomar la iniciativa cuando encuentre una etiqueta RFID vacía y consulte al usuario por la información del lugar y sus conexiones.



**Fig. 7.34:** Primer caso de señalización: El robot toma la iniciativa en la interacción.

<sup>1</sup> Guía Práctica del Campus <http://goo.gl/CxbnZ>

### ¿Dónde estoy? [dialogPlaceWaymarking]

**Diálogo 1** En el primer ejemplo de diálogo realizado, el robot hace una consulta simple acerca de su ubicación, preguntando por los elementos básicos de un *lugar*: el nombre y su identificador.

#### *Diálogo 7.3: Solicitud de información del lugar actual*

- |    |  |
|----|--|
| 1  | <b>Robot:</b> ¿Cómo se llama este lugar?   |
| 3  | <b>Usuario:</b> Este es el Laboratorio de Robótica   |
| 5  | <b>Robot:</b> Ok, dame mas información sobre este sitio.<br>¿Cuál es el identificador de este lugar? |
| 7  |  |
| 9  | <b>Usuario:</b> Es el 1 punto 3 C 14   |
| 11 | <b>Robot:</b> ¿Tienes mas información de este lugar?   |
| 13 | <b>Usuario:</b> No, no la tengo.   |
|    | <b>Robot:</b> Muy bien, gracias por tu ayuda.  |

El diálogo mostrado especifica una manera coloquial de pedir información para que el robot pueda almacenarla en una señal. El robot inicialmente pregunta por el nombre del lugar donde se encuentra. La respuesta que recibe del usuario, se corresponde con uno de los nombres de los posibles lugares a donde puede ir el robot, especificados previamente en la gramática. Posteriormente, el robot pregunta por el identificador del lugar y consulta al usuario por información adicional, en este ejemplo, el usuario responde al robot que no posee más información y se termina el proceso de adquisición de información del lugar actual.

**Diálogo 2** En el siguiente diálogo, el usuario proporciona al robot más información sobre el lugar en el que se encuentra, es decir, proporciona información de las *zonas anidadas*.

#### *Diálogo 7.4: Solicitud de información del lugar actual con zonas anidadas*

- |    |  |
|----|--|
| 1  | <b>Robot:</b> ¿Cómo se llama este lugar?   |
| 3  | <b>Usuario:</b> Éste es el Laboratorio de Robótica   |
| 5  | <b>Robot:</b> Ok, dame mas información sobre este sitio.<br>¿Cuál es el identificador de este lugar? |
| 7  |  |
| 9  | <b>Usuario:</b> Es el 1 punto 3 C catorce  |
| 11 | <b>Robot:</b> ¿Tienes mas información de este lugar?   |
| 13 | <b>Usuario:</b> Si.  |
|    | <b>Robot:</b> Me has dicho que tienes más datos. Dame más información sobre este sitio.              |



15 **Usuario:** Estamos en el Edificio Betancourt.  
17 **Robot:** ¿Cuál es el identificador de éste lugar?  
19 **Usuario:** Es el 1  
21 **Robot:** ¿Tienes mas información de este lugar?  
23 **Usuario:** Si.  
25 **Robot:** Me has dicho que tienes más datos. Dame más información sobre este sitio.  
27 **Usuario:** Estamos en la Tercera Planta.  
29 **Robot:** ¿Cuál es el identificador de éste lugar?  
31 **Usuario:** Es el 1 punto 3  
33 **Robot:** ¿Tienes mas información de este lugar?  
35 **Usuario:** No, no la tengo.  
37 **Robot:** Muy bien, gracias por tu ayuda.

### Formularios utilizados – Diálogo en voiceXML

A continuación se presenta el formulario en **voiceXML** encargado de la adquisición de información del lugar donde se encuentra el robot (**dialogPlaceWaymarking**).

- Primeramente, el intérprete debe obtener la gramática para el reconocimiento automático (línea 31).
- Una vez que se ha cargado la gramática, el robot procede a preguntar al usuario los datos del lugar actual, mediante el campo **lugares** del formulario (línea 38). Como cada lugar está conformado por zonas anidadas, el robot puede seguir preguntado por más información acerca de ese lugar (línea 40). Además del nombre del lugar, también pregunta por su identificador (línea 44).
- Cuando el sistema de diálogo completa la información del lugar donde se encuentra la señal, los datos se colocan en la Memoria a Corto Plazo del sistema y se emite un evento, notificando que se han almacenado datos de un lugar (líneas 54 a 59).
- Si el usuario le ha indicado al robot que posee más información del lugar (es decir, completar la información de zonas anidadas del lugar), el intérprete de diálogos llama nuevamente al formulario **dialogPlaceWaymarking** (líneas 61 a 64).

*Formulario para la adquisición de información de un lugar*

```

30 <form id="dialogPlaceWaymarking">
    <block>
        <prompt count="1">#setGrammar$waymarking.gram</prompt>
32     <assign name="EVENT_PLACE" expr="500102"/>
    </block>
34 <grammar src="../../Grammars/trivial.grxml" type="application/srgs+xml"
        mode="voice" xml:lang="es-ES"
36     version="1.0"/>

38 <field name="lugares">
    <prompt cond="info == 1">Como se llama este lugar?</prompt>
40 <prompt cond="info == 0">Dame mas informacion sobre este sitio</prompt>
    <filled><assign name="place_label" expr="lugares"/></filled>
42 </field>
    <field name="ids">
44 <prompt>cual es el identificador de este lugar?</prompt>
    <filled><assign name="place_ident" expr="ids"/></filled>
46 </field>
    <field name="op">
48 <prompt>Tienes mas informacion de este lugar?</prompt>
    <filled>
50 <if cond="op == 'si'"><assign name="info" expr="'0'"/>
    <else/><assign name="info" expr="'1'"/></if>
52 </filled>
    </field>
54 <filled><var name="place"
        expr="addStrings(place_label,place_ident,info)"/>
56 <prompt>#ColocarDato$<value expr="place"/></prompt>
    <prompt>#emit$<value expr="EVENT_PLACE"/></prompt>
58 <prompt> Vale</prompt>
    </filled>
60 <block>
    <if cond="info == 0">
62 <prompt> Me has dicho que tienes mas datos </prompt>
    <goto next="#dialogPlaceWaymarking"/>
64 <else/><goto next="#dialogOptionWaymarking"/></if>
    </block>
66 </form>

```

### Solicitud de información adicional [dialogOptionWaymarking]

Este diálogo consulta al usuario si posee información adicional para la señalización. Su finalidad es servir como conexión entre el diálogo del tipo *¿Dónde estoy?* y el diálogo del tipo *¿Hacia dónde puedo ir?*. Pueden haber dos casos, dependiendo de la respuesta del usuario, si es positiva, se activará el siguiente diálogo (para obtener información de las conexiones) y si es negativa, el diálogo finaliza.

***Diálogo 7.5: Solicitud de confirmación de información adicional***

**Robot:** ¿Desde aquí puedo ir a otros sitios?

**Usuario:** Si Maggie.

**Robot:** Muy bien, estoy atenta a tus indicaciones.

**Robot:** ¿Desde aquí puedo ir a otros sitios?

**Usuario:** No.

**Robot:** Okey, gracias por tu ayuda.

A continuación se presenta el formulario en voiceXML encargado de solicitar información adicional (`dialogOptionWaymarking`).

***Consulta al usuario si posee información de las conexiones***

```

68 <form id="dialogOptionWaymarking">
    <block>
        <prompt count="1">#setGrammar$waymarking.gram</prompt>
70 </block>
        <grammar src="../../../Grammars/trivial.grxml" type="application/srgs+xml"
72         mode="voice" xml:lang="es-ES" version="1.0"/>
        <field name="op">
74         <prompt>Desde aqui puedo ir a otros sitios?</prompt>
            <filled>
76             <if cond="op=='no'">
                <prompt> Okey gracias por tu ayuda </prompt>
78                 <goto next="#dialogSaveWaymarking"/>
            <else/>
80                 <prompt> Muy bien, estoy atenta a tus indicaciones</prompt>
                <goto next="#dialogConnectionWaymarking"/></if>
82             </filled>
        </field>
84 </form>

```

***¿Hacia donde puedo ir? [dialogConnectionWaymarking]***

El objetivo de este diálogo es que el robot pueda adquirir la información de las conexiones, o lugares hacia donde puede ir, desde el sitio donde se encuentra la señal que se quiere escribir o actualizar. El usuario podrá darle el número de conexiones que conozca.

Para el siguiente ejemplo de diálogo, el usuario indica sólo una conexión. El robot durante el diálogo solicita los datos de uno de los elementos claves de la conexión: el *lugar*.

***Diálogo 7.6: Solicitud de información de una conexión. Información de un lugar.***

2 **Robot:** Dime, ¿hacia donde puedo ir?

4 **Usuario:** Puedes ir al Laboratorio de Robótica 2

6 **Robot:** ¿Cuál es el identificador de ese lugar?

8 **Usuario:** Es el 1 punto 3 C 13

10 **Robot:** ¿Tienes más información del lugar?

12 **Usuario:** No Maggie

14 **Robot:** Okey.

Como se explicó en la sección 3.3.2, una *conexión* está conformada por un *lugar* y una *acción* (o lista de *acciones*). Al igual que en los diálogos comentados en 7.3.2, el usuario puede indicar al robot las zonas anidadas del lugar hacia donde se puede ir desde el sitio que se está señalizando. A continuación se muestra un ejemplo de éste tipo de diálogo.

***Diálogo 7.7: Solicitud de información de una conexión. Información de un lugar con zonas anidadas.***

2 **Robot:** Dime, ¿hacia donde puedo ir?

4 **Usuario:** Puedes ir al Laboratorio de Robótica 2

6 **Robot:** ¿Cuál es el identificador de ese lugar?

8 **Usuario:** Es el 1 punto 3 C 13

10 **Robot:** ¿Tienes más información del lugar?

12 **Usuario:** Si

14 **Robot:** Dame más información sobre ese sitio.

16 **Usuario:** Estamos en el Edificio Betancourt.

18 **Robot:** ¿Cuál es el identificador de éste lugar?

20 **Usuario:** Es el 1

22 **Robot:** ¿Tienes mas información de este lugar?

24 **Usuario:** Si.

26 **Robot:** Dame más información sobre este sitio.

28 **Usuario:** Estamos en la Tercera Planta.  
30 **Robot:** ¿Cuál es el identificador del lugar?  
32 **Usuario:** Es el 1 punto 3  
34 **Robot:** ¿Tienes mas información de ese lugar?  
36 **Usuario:** No Maggie.  
38 **Robot:** Okey.

Una vez que se ha completado la información correspondiente al *lugar* de una *conexión*, el robot procede a consultar la información de la *acción* o conjunto de acciones que se debe realizar para ir al *lugar* previamente indicado. A continuación se muestra un ejemplo de este tipo de diálogo.

***Diálogo 7.8: Solicitud de información de una conexión. Información de una acción.***

**Robot:** ¿Qué debo hacer?  
2  
**Usuario:** Para ir ahí, debes seguir por la derecha.  
4  
**Robot:** ¿Crees que necesite un mapa geométrico?  
6  
**Usuario:** No Maggie.  
8  
**Robot:** Okey.  
10  
**Robot:** Hay otro sitio al que pueda ir desde aquí?  
12  
**Usuario:** No, no conozco ningún otro sitio.  
14  
**Robot:** Muy bien, gracias por tu ayuda.  
16       Voy a almacenar la información.

Adicionalmente, en el diálogo, una vez que se complete la información de una conexión, el robot puede continuar consultando al usuario por otras conexiones, es decir, por otros lugares a donde se puede ir desde la señal que se está escribiendo. por lo que el siguiente diálogo, pregunta al usuario por la existencia de éstos. Si la respuesta es afirmativa, se repite el mismo patrón de los diálogos vistos anteriormente.

***Diálogo 7.9: Consulta por otras conexiones.***

2 **Robot:** Hay otro sitio al que pueda ir desde aquí?  
4 **Usuario:** Si Maggie.  
6 **Robot:** Me has dicho que puedo ir a otro lugar.

## Formularios utilizados – Diálogo en voiceXML

A continuación se presenta la estructura en VXML del diálogo `dialogConnectionWaymarking`.

- Inicialmente se declaran un conjunto de constantes que determinan los eventos que notifican cuando se ha adquirido la información de una conexión, como el nombre del lugar o lugares a donde el robot puede ir desde la señal establecida y la acción que debe ejecutar (líneas 88 a 91).

El robot pregunta por el nombre de la conexión y su identificador (líneas 97 y 104), y por los datos que corresponden a las zonas anidadas de la conexión. Una vez que haya adquirido éste primer conjunto de datos, los envía a la Memoria a Corto Plazo del sistema y emite el evento `EVENT_CONNECTION` (líneas 120-121).

### *Formulario para obtener información de una conexión*

```

86 <form id="dialogConnectionWaymarking">
87   <block>
88     <prompt count="1">#setGrammar$waymarking.gram</prompt>
89     <assign name="EVENT_CONNECTION" expr="500103"/>
90     <assign name="EVENT_ACTION" expr="500106"/>
91     <assign name="EVENT_ENDING_CONNECTION" expr="500113"/>
92     <assign name="EVENT_NO_ENDING_CONNECTION" expr="500114"/>
93     <assign name="info_a" expr="1"/><assign name="indicaciones" expr="1"/>
94   </block>
95   <grammar src="../../Grammars/trivial.grxml" type="application/srgs+xml"
96     mode="voice" xml:lang="es-ES" version="1.0"/>
97   <field name="lugares">
98     <prompt cond="info_c == 1">Dime, hacia donde puedo ir?</prompt>
99     <prompt cond="info_c == 0">Dame mas informacion sobre ese sitio</prompt>
100    <filled>
101      <assign name="connection_label" expr="lugares"/>
102    </filled>
103  </field>
104  <field name="ids">
105    <prompt>cual es el identificador de ese lugar?</prompt>
106    <filled><assign name="connection_id" expr="ids"/></filled>
107  </field>
108  <field name="op">
109    <prompt>Tienes mas informacion del lugar?</prompt>
110    <filled>
111      <if cond="op == 'si'">
112        <assign name="info_c" expr="'0'"/>
113      <else/>
114        <assign name="info_c" expr="'1'"/>
115      </if>
116    </filled>
117  </field>
118  <block>
119    <var name="lugar_conexion"
120      expr="addStrings(connection_label,connection_id,info_c)"/>

```

```

120 <prompt>#ColocarDato$<value expr="lugar_conexion"/></prompt>
    <prompt>#emit$<value expr="EVENT_CONNECTION"/></prompt>
122 <prompt> Okey </prompt>
    </block>

```

- En la siguiente parte del diálogo, el robot pregunta por la acción que debe realizar (línea 131). Como la acción puede o no ser geométrica, el robot consulta al usuario si para ejecutar la acción indicada requiere un mapa geométrico (línea 135).

*Consulta al usuario por las acciones*

```

124 <block>
    <if cond="info_c == 0">
126     <prompt> Me has dicho que tienes mas datos </prompt>
        <goto next="#dialogConnectionWaymarking"/>
128    </if>
    </block>
130    <field name="acciones">
        <prompt>Que debo hacer?</prompt>
132        <filled><assign name="action_label" expr="acciones"/></filled>
    </field>
134    <subdialog name="tipodeaccion" src="response_yes_no.vxml">
        <prompt> Crees que necesite un mapa geometrico? </prompt>
136        <param name="myquestion" expr="Necesito un mapa geometrico?"/>
    </subdialog>
138    <filled mode="all" namelist="tipodeaccion">
        <if cond="tipodeaccion.sino=='no'">
140            <prompt> Okey </prompt>
            <assign name="tipo_action" expr="'0'"/>
142        <else/>
            <prompt> Okey </prompt>
144            <assign name="tipo_action" expr="'1'"/>
        </if>
146    </filled>
    <filled>
148        <assign name="code_action" expr="getCode(action_label)"/>
    </filled>
150    <filled>
        <var name="accion_conexion"
152            expr="addStrings2(action_label,tipo_action,code_action,info_a)"/>
        <prompt>#ColocarDato$<value expr="accion_conexion"/></prompt>
154        <prompt>#emit$<value expr="EVENT_ACTION"/></prompt>
    </filled>

```

- Al completarse la información de una conexión, el robot pregunta por la existencia de otras conexiones. Si el usuario indica que no hay más conexiones, el sistema emite el evento `EVENT_ENDING_CONNECTION` para notificar que ya se han obtenido los datos de las conexiones para ese lugar, y se guardan los datos en la Memoria Corto Plazo llamando a `dialogSaveWaymarking`. En caso

que el usuario indique la existencia de otras conexiones, el sistema de diálogos emitirá el evento `EVENT_NO_ENDING_CONNECTION`, y volverá al formulario `dialogConnectionWaymarking` para completar la información de las siguientes conexiones.

### *Consulta al usuario por las acciones*

```

156 <subdialog name="otra_conexion" src="response_yes_no.vxml">
158   <prompt> Hay otro sitio al que pueda ir desde aqui?</prompt>
      <param name="myquestion" expr="Hay otro sitio al que pueda ir desde aqui?"/>
160 </subdialog>

162 <filled mode="all" namelist="otra_conexion">
      <if cond="otra_conexion.sino=='no'">
164   <prompt> Muchas gracias por la informacion </prompt>
      <assign name="info_a" expr="'1'"/>
166   <prompt>#emit$<value expr="EVENT_ENDING_CONNECTION"/></prompt>
      <else/>
168   <prompt> Me has dicho que puedo ir a otro lugar </prompt>
      <assign name="info_a" expr="'0'"/>
170   <prompt>#emit$<value expr="EVENT_NO_ENDING_CONNECTION"/$></prompt>
      <goto next="#dialogConnectionWaymarking"/>
172   </if>
</filled>
174 <filled>
      <if cond="info_a=='1'">
176   <goto next="#dialogSaveWaymarking"/>
      </if>
178 </filled>
</form>

```

### 7.3.3. El usuario toma la iniciativa

En los diálogos anteriores el robot ejecuta los diálogos cuando se encuentra al lado de una señal (etiqueta RFID) y toma la iniciativa en la interacción humano-robot para completar la información de la señal. Uno de los objetivos en la tarea de *Señalización* consiste en facilitar la escritura y modificación de las señales a los usuarios, es aquí donde el papel de la interacción humano-robot toma importancia, el usuario toma la iniciativa al proporcionarle la información referente a la señal que se quiera escribir para posteriormente guiarle por el entorno para alcanzar otras etiquetas.

#### El usuario indica al robot el lugar actual

En el siguiente diálogo el usuario indica al robot el lugar donde se encuentra (nombre e identificador). El lugar no posee zonas anidadas.



***Diálogo 7.10: El usuario toma la iniciativa para dar información al robot.***

2 **Usuario:** Maggie, voy a darte las indicaciones de este lugar.  
Este lugar es el Laboratorio de Robótica 3.

4 **Robot:** Entendido.

6 **Usuario:** El identificador de este lugar es el 1 punto 3 c 12

8 **Robot:** Muy bien, ¿tienes más información de este lugar?

10 **Usuario:** No, no la tengo.

12 **Robot:** Entonces guardaré la información.

De modo similar a como se ha visto en el diálogo 7.4, el usuario podrá dar información al robot de las zonas anidadas, con la diferencia que el robot no solicita la información sino espera a que el usuario se la indique.

**El usuario indica al robot acciones de movimiento**

Como se ha dicho en el capítulo 5, la importancia de que el robot pueda señalar el entorno (escribir información en las etiquetas RFID), está relacionada directamente con la automatización del sistema. El usuario podrá interactuar con el robot con un diálogo coloquial y “*enseñarle el entorno*” al robot, para que éste pueda almacenar la información en etiquetas distribuidas por el entorno.

El diálogo que se realizará para éste fin, será como el que se muestra a continuación.

***Diálogo 7.11: El usuario indica al robot acciones de movimiento.***

2 **Usuario:** Maggie, sígueme.

4 **Robot:** Muy bien.

6 **Robot:** He detectado una etiqueta.

8 **Usuario:** Muy bien Maggie. Voy a darte las indicaciones de este lugar.

10 ....

12 **Robot:** Entonces guardaré la información.

14 **Usuario:** Ahora sígueme, seguiré mostrándote el lugar.

16 **Robot:** Muy bien.

18 **Usuario:** Maggie, voy a darte las indicaciones de este lugar.

20 Este lugar es...

## Gramática y formularios utilizados – Diálogos en voiceXML

Para estos experimentos se modificó la gramática utilizada previamente (sección 7.3.1). A diferencia de la anterior se ha añadido una nueva regla gramatical denominada \$indicaciones (línea 37) y se han añadido otras acciones de movimiento en la regla \$acciones (línea 26).

```

#ABNF 1.0 ISO-8859-1;
2
language es-ES;
4 tag-format <loq-semantic/1.0>;
6
public $root = $waymarking_guide;
8
public $waymarking_guide = $adjetivos | $ids | $op | $acciones |
    $codigos | $tipodeaccion | $indicaciones;
10
public $lugares = ("edificio betancourt" | "tercera planta" |
12 "departamento de ingeniería de sistemas y automática zona c" |
    "laboratorio de maggie" | "laboratorio de robotica 2" |
14 "departamento de ingeniería de sistemas y automática zona b" |
    "departamento de ingeniería eléctrica" |
16 "laboratorio de robótica 3" | "servicios" | "televisión" | "tele" |
    "tercera planta" | "zona" | "despacho" ){<@lugares $value>};
18
public $ids = ("uno" | "1 punto 3" | "1 punto 3 c" | "1 punto 3 c 12"
20 | "1 punto 3 c 13" | "1 punto 3 b" | "1 punto 3 d" | "1 punto 3 c 14"
    | "1 punto 3 c s" | "1 punto 3 c 12 tv" ){<@ids $value>};
22
public $op = ("si":si | "no":no | "okey":si | "vale":si | "negativo":no
24 | "de acuerdo":si | "incorrecto":no ){<@op $value>};
26
public $acciones = ("sigue el contorno por la derecha"
    | "sigue el contorno por la izquierda"
28 | "usa la habilidad ir a un punto" | "entrar" | "salir" |
    "sigueme" | "ven"){<@acciones $value>};
30
public $adjetivos =(("esto es"|"estas en"|"aquí es"|"puedes ir a"|"hacia"))$lugares;
32
public $codigos = ("0" | "1" | "2" | "3"){<@codigos $value>};
34
public $tipodeaccion = ("si" | "no"){<@tipodeaccion $value>};
36
public $indicaciones = ("indicaciones" | "lugar" | "identificador"
38 | "puedes" | "ir"){<@indicaciones $value>};

```

A continuación se presenta la estructura en VXML del diálogo dialogWaymarking-Guide.

- El diálogo es similar al visto en la sección 7.3.2, con la diferencia de que en los

formularios, el robot no pregunta por la información del lugar, sino espera las indicaciones del usuario (líneas 38 y 42).

- Cuando el sistema de diálogo completa la información del lugar donde se encuentra la señal, los datos se colocan en la Memoria a Corto Plazo del sistema y se emite un evento, notificando que se han almacenado datos de un lugar (líneas 54 a 56).
- El robot pregunta al usuario si posee más información del lugar, si el usuario responde afirmativamente (es decir, posee información de zonas anidadas del lugar), el intérprete de diálogos llama nuevamente al formulario dialog-WaymarkingGuide (líneas 60 a 64).

#### *Formulario para la indicación de información de un lugar*

```

30 <form id="dialogWaymarkingGuide">
    <block>
        <prompt count="1">#setGrammar$waymarking_guide.gram</prompt>
32     <assign name="EVENT_PLACE" expr="500102"/>
    </block>
34 <grammar src="../../Grammars/trivial.grxml" type="application/srgs+xml"
        mode="voice" xml:lang="es-ES"
36     version="1.0"/>

38 <field name="lugares">
    <filled><assign name="place_label" expr="lugares"/></filled>
40 <prompt>Entendido</prompt>
</field>
42 <field name="ids">
    <filled><assign name="place_ident" expr="ids"/></filled>
44 <prompt>Muy bien</prompt>
</field>
46 <field name="op">
    <prompt>Tienes mas informacion de este lugar?</prompt>
48 <filled>
    <if cond="op == 'si'"><assign name="info" expr="'0'"/>
50 <else/><assign name="info" expr="'1'"/></if>
    </filled>
52 </field>
<filled><var name="place"
54     expr="addStrings(place_label,place_ident,info)"/>
    <prompt>#ColocarDato$<value expr="place"/></prompt>
56 <prompt>#emit$<value expr="EVENT_PLACE"/></prompt>
    <prompt> Entonces guardaré la información</prompt>
58 </filled>
    <block>
60 <if cond="info == 0">
        <prompt> Me has dicho que tienes mas datos </prompt>
62 <goto next="#dialogPlaceWaymarking"/>
        <else/><goto next="#dialogOptionWaymarking"/></if>
64 </block>
</form>

```

Como se ha dicho anteriormente, el robot además de estar atento a la información referente al contenido de las señales, también debe estar atento a las indicaciones de movimiento. El siguiente diálogo en VXML, espera las órdenes de movimiento de parte del usuario. El diálogo se ha hecho de manera similar a cuando el robot solicita las acciones que debe realizar para ir a un lugar, con la diferencia que espera las indicaciones del usuario. Una vez recibida la acción de movimiento que le ordena el usuario, envía esta información a la Memoria a Corto Plazo del sistema (línea 34) para que un Manejador de Acciones envíe la orden de movimiento a la habilidad correspondiente.

*Formulario para dar órdenes de movimiento*

```

2   <form id="dialogMotionCommands">
    <block>
        <prompt count="1">#setGrammar$waymarking_motionCommands.gram</prompt>
4       <assign name="EVENT_MOTION" expr="500121"/>
    </block>
6   <grammar src=" ../Grammars/trivial.grxml" type="application/srgs+xml"
        mode="voice" xml:lang="es-ES"
8       version="1.0"/>

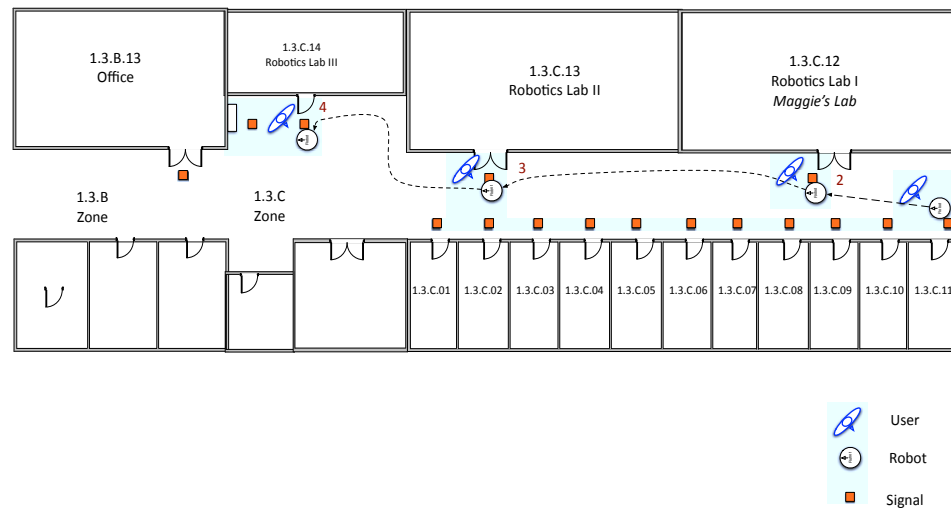
10  <field name="movimiento">
        <filled><assign name="motion_label" expr="acciones"/></filled>
12    <prompt>Muy bien</prompt>
    </field>

14  <subdialog name="tipodeaccion" src="response_yes_no.vxml">
16    <prompt> Crees que necesite un mapa geometrico? </prompt>
    <param name="myquestion" expr="Necesito un mapa geometrico?"/>
18  </subdialog>
    <filled mode="all" namelist="tipodeaccion">
20    <if cond="tipodeaccion.sino=='no'">
        <prompt> Okey </prompt>
22    <assign name="tipo_action" expr="''0'"/>
    <else/>
24    <prompt> Okey </prompt>
        <assign name="tipo_action" expr="''1'"/>
26    </if>
    </filled>
28  <filled>
        <assign name="code_action" expr="getCode(action_label)"/>
30  </filled>
    <filled>
32    <var name="accion_conexion"
        expr="addStrings2(action_label,tipo_action,code_action,info_a)"/>
34    <prompt>#ColocarDato$<value expr="accion_conexion"/></prompt>
        <prompt>#emit$<value expr="EVENT_MOTION"/></prompt>
36  </filled>
38  </form>

```

En la figura 7.35, se muestra un ejemplo de señalización donde el usuario da órdenes de movimiento al robot para ir señalizando algunas etiquetas de la Tercera Planta del Edificio Betancourt de la UC3M.

1. El usuario comienza el recorrido con el robot en la señal 1. Le indica al robot la información del lugar *Despacho-1.3.C.11*. Posteriormente, da al robot la orden de movimiento “*Sígueme*” para llegar a la señal 2.
2. En la señal 2, el usuario indica al robot la información del lugar “*Laboratorio de Maggie-1.3.C.12*”. El robot almacena la información y espera una nueva orden del usuario. El usuario ordena al robot “*seguir el contorno por la derecha*”.
3. En la señal 3, el usuario indica al robot la información del lugar “*Laboratorio de Robótica 2-1.3.C.13*”. El robot almacena la información y espera una nueva orden del usuario. El usuario ordena al robot “*seguir el contorno por la derecha*”.
4. Para finalizar, cuando el robot detecta la señal 4, espera a que el usuario le proporcione la información del lugar “*Laboratorio de Robótica 3-1.3.C.14*”, cuando el usuario termina de darle los datos el robot almacena la información, el robot espera la notificación del usuario que ya se ha terminado de señalizar el entorno y se finaliza la ejecución de la habilidad.



**Fig. 7.35:** Ejemplo de señalización del entorno. El usuario da órdenes de movimiento al robot.

#### 7.3.4. Escritura de la información

Como se explicó previamente en la sección 6.8, la Habilidad de Señalización (CSkill\_Waymarking), es la habilidad diseñada y desarrollada en esta tesis, encargada de gestionar la información adquirida a través de la interacción humano-robot y

darle el formato adecuado de una señal, para posteriormente enviarla a la habilidad de escritura de señales. A medida que el diálogo se está desarrollando, es esta habilidad la que estructura la información que será escrita en la señal. Una vez que el proceso de la habilidad da formato a la información recibida desde el gestor de diálogo, los datos de la señal se envían a la Memoria a Corto Plazo del sistema y se emite un evento que notifica a la habilidad de escritura de etiquetas RFID que hay datos disponibles para escribir físicamente la señal.

La Habilidad de Señalización estructura los datos adquiridos en la interacción humano-robot, en un fichero XML, cada señal tendrá escrita físicamente ésta información. Así para los ejemplos de diálogos presentados anteriormente, la información queda de la siguiente manera:

*Información del lugar a señalar.*

```

1 <?xml version="1.0"?>
2 <Node_Navigation>
3   <NodeID></NodeID>
4   <place>
5     <zone>
6       <label>Laboratorio de Maggie</label>
7       <ID>1.3.C.12</ID>
8     </zone>
9     <zone>
10      <label>Zona C</label>
11      <ID>1.3.C</ID>
12    </zone>
13    <zone>
14      <label>Tercera Planta</label>
15      <ID>1.3</ID>
16    </zone>
17    <zone>
18      <label>Edificio Betancourt</label>
19      <ID>1</ID>
20    </zone>
21  </place>
22 </Node_Navigation>

```

*Información de las conexiones.*

```

1 <?xml version="1.0"?>
2 <Node_Navigation>
3   <NodeID></NodeID>
4   <place />
5   <connection>
6     <place>
7       <zone>
8         <label>Laboratorio de Robotica II</label>
9         <ID>1.3.C.13</ID>
10      </zone>
11    </place>

```

```

12         <action>
13             <nombre>Avanzar por la derecha</nombre>
14             <tipo>0</tipo>
15             <codigo>2</codigo>
16         </action>
17     </connection>
18     <connection>
19         <place>
20             <zone>
21                 <label>Departamento de Automática zona b</label>
22                 <ID>1.3.B</ID>
23             </zone>
24         </place>
25         <action>
26             <nombre>Avanzar por la izquierda</nombre>
27             <tipo>0</tipo>
28             <codigo>1</codigo>
29         </action>
30     </connection>
</Node_Navigation>

```

## 7.4. Discusión

En la primera parte de los experimentos se ha podido comprobar que el robot detecta y lee correctamente las señales ubicadas en un entorno interior y estructurado. El robot ejecuta correctamente las acciones indicadas en las señales para llegar a la meta y alcanza el objetivo sin problemas. También se ha probado colocar un obstáculo móvil en una de las señales, específicamente en la señal que indicaba como lugar actual el destino al que debía llegar el robot, al estar el entorno señalizado, el robot pudo comprobar en la siguiente señal que debía cambiar de sentido, siguiendo las indicaciones de la señal, el robot alcanzó satisfactoriamente la meta. En los siguientes experimentos el robot no conocía a priori su ubicación, por lo que la primera fase de la navegación consistió en explorar el entorno buscando una señal que le indicase su ubicación, una vez ubicado, el robot prosiguió a leer la información de las señales y, siguiendo las indicaciones, logró llegar a la meta establecida. Podemos observar que este comportamiento es similar al que hacemos las personas cuando nos encontramos en un entorno desconocido, inicialmente buscamos una señal que nos indique nuestra ubicación para posteriormente buscar una indicación que nos guíe a nuestra meta. Este tipo de comportamiento natural es uno de los objetivos que se ha propuesto en esta tesis.

El sistema de navegación propuesto en esta tesis, no solo considera la navegación topológica, sino también toma en cuenta el uso de la navegación geométrica. En el experimento realizado se ha utilizado este tipo de navegación cuando el robot necesita llegar a un punto geométrico concreto, en el experimento, el robot tenía que ubicarse justo enfrente de la televisión y orientado hacia ésta para tener alcance al receptor infrarrojo y ejecutar correctamente la orden “*Apagar la Televisión*”.

En la segunda parte de los experimentos se probó la interacción humano-robot

mediante diálogos para que el robot, con la información que le proporcione una persona, pudiese crear o modificar la información contenida en una señal. Las pruebas realizadas comprueban que la naturalidad con la que el robot solicita la información o como la persona se la comunica, hace que el proceso de escritura de las señales sea mas cómodo sin necesidad de que el usuario tenga conocimientos técnicos en tecnología RFID. El sistema realizado permite que tanto el humano como el robot puedan tomar la iniciativa en el diálogo, el robot usa frases comunes del tipo *¿Dónde estoy?* y *¿hacia dónde puedo ir?*, así como lo hacemos los humanos, cuando preguntamos a otras personas por información del entorno. Igualmente, es posible guiar al robot e ir “mostrándole” el entorno para que vaya realizando la tarea de escritura de las señales.



## 8. CONCLUSIONES Y TRABAJOS FUTUROS

*“Detrás de lo que creemos conocer de sobra se esconde una cantidad equivalente de desconocimiento.”*  
(Haruki Murakami)

La idea de tener un robot interactuando con el ser humano en casa o en espacios públicos, desenvolviéndose por el entorno de la misma manera que una persona, no es un problema simple. La navegación autónoma de un robot es un objetivo fundamental que ha sido y será objeto de numerosas investigaciones en el campo de la robótica. Es una acción natural en las personas, que al visitar por primera vez un sitio y querer llegar a un lugar específico, usamos los recursos del entorno para orientarnos y buscar el modo correcto de llegar a nuestro destino. Es por eso que en esta tesis se ha usado la señalización en el entorno como un soporte para lograr con éxito la navegación autónoma de un robot, tratando de emular así el proceso cognitivo que usamos los humanos.

En la actualidad, existen pocos trabajos que enfoquen la tarea de navegación en robótica de manera natural y que, tal y como lo hacemos los humanos, usen la información espacial del entorno como recurso básico para la navegación. Uno de los principales aspectos que influyen en el rendimiento de estos sistemas es el tipo de sensores que se usan para adquirir la información. La mayoría de los trabajos relacionados con esta tesis, usan sistemas de visión para leer la información de marcas naturales del entorno, con el inconveniente de depender de la luz para su correcto funcionamiento. Es por ese motivo que para esta tesis, debido a la tecnología existente en la actualidad, se ha escogido utilizar los sistemas de Identificación por Radio Frecuencia (RFID), ya que son un buen recurso para enriquecer la información que un robot puede obtener del entorno, acelerando considerablemente el proceso de lectura de las señales sin depender de factores ambientales como la luz.

Otro de los factores importantes en este tipo de navegación es la señal en sí. La señales que usamos son etiquetas RFID pasivas, para uso exclusivo del robot (o los robots), estas señales son de bajo coste, no perjudican la transitabilidad de las personas ni la de los robots y no ejercen un impacto al entorno, ya que como hemos visto, para la navegación es posible colocarlas en la pared y pintarlas del mismo color, de manera que no afecte visualmente al entorno. La facilidad de la escritura de las etiquetas RFID también fueron un punto a favor para escoger esta tecnología, las habilidades de escritura y lectura desarrolladas para esta tesis, facilitan la escritura de la información de las señales para navegación.

En cuanto a la información que almacenan, con esta tecnología se pueden almacenar datos más enriquecedores para la orientación del robot, en cada señal el robot no solo puede obtener información del lugar en el que se encuentra sino también los posibles lugares a los que puede ir o si esos lugares le pueden llevar al destino deseado. Por otra parte, en este sistema se ha añadido no sólo información topológica, sino que es posible manejar datos geométricos, como posiciones en un mapa geométrico concreto y acciones de movimiento que implican precisión, lo que hace a este sistema mas completo a la hora de realizar las tareas de navegación.

Adicionalmente, el hecho de que la implementación de este sistema de navegación sea en un robot social, nos ha motivado a incluir la interacción humano-robot como un factor que facilita la tarea de señalización del entorno.

Finalmente, al ver los objetivos que se habían trazado inicialmente al comienzo del desarrollo de esta tesis, podemos destacar que se han alcanzado satisfactoriamente, hecho que se ha comprobado al implementar y experimentar con el sistema en un robot social.

En las siguientes secciones se presentarán las principales contribuciones en el campo de la investigación de la robótica que se han logrado con esta tesis, además de los posibles trabajos de investigación y líneas futuras que se abren al finalizar este trabajo.

## 8.1. Principales Aportaciones

Las aportaciones de esta tesis son principalmente dos:

1. Se ha propuesto un **sistema de navegación mediante señales**, basado en el comportamiento de los seres humanos, de manera tal que un robot móvil sin tener conocimiento del entorno y sin un mapa previo, pueda llegar a un destino guiándose mediante las señales que encuentra por el entorno.

Esta aportación incluye:

- El diseño del sistema de señalización y el modelado de la información que almacenan las señales. Se incluyeron en el modelo, conceptos importantes como *Lugar*, *Conexión* y *Acción*.
- La integración de sensores RFID en el robot y el desarrollo del software para las operaciones de lectura y escritura de etiquetas.
- El diseño y desarrollo del algoritmo de navegación que usa como entrada un lugar destino y la información que lee el robot de las señales a medida que navega. El algoritmo analiza la información y coordina las acciones de movimiento que debe realizar el robot para alcanzar satisfactoriamente su objetivo.
- El sistema de navegación maneja información geométrica para los casos que sean necesarios, por ejemplo: entornos no estructurados como una habitación. El sistema es capaz de usar un mapa geométrico del entorno que la señal le indique, localizarse geométricamente y ejecutar habilidades que usen esta información (como la habilidad *Ir a un punto*).

2. Se ha propuesto un **sistema de escritura de las señales del entorno**, de modo que el robot pueda interactuar con las personas para adquirir la información de las señales que se quieren escribir y así automatizar el proceso de señalización.

Esta aportación incluye:

- El diseño del sistema de diálogo para adquirir la información de la señal a escribir. Para esto se ha considerado que tanto el humano como el robot puedan tomar la iniciativa en el diálogo, es decir, que el robot pueda consultar al usuario la información de una señal con preguntas del tipo *¿Dónde estoy?* o *¿Hacia dónde puedo ir?* o que una persona pueda guiar al robot a través del entorno para señalizarlo, usando frases comunes como *“Esto es...”*, además de darle órdenes de movimiento como *“Ven, sígueme...”*.
- El diseño y desarrollo del algoritmo de señalización que interpreta la información, obtenida a través del diálogo, y le da el formato lógico de una *señal*. Este algoritmo también envía ésta información físicamente a la señal a través de la habilidad de escritura de RFID, además también coordina las órdenes de movimiento que una persona ordena al robot.

Los sistemas descritos fueron integrados en la arquitectura de software Automático - Deliberativa (AD) en forma de *habilidades*, y además se han implementado y experimentado con éxito en el robot social Maggie.

## 8.2. Trabajos Futuros

Aparte de las posibles mejoras que se pueden hacer al sistema de navegación, uno de los objetivos adicionales al hacer una tesis doctoral es que se abran líneas futuras de investigación a partir del trabajo realizado. A continuación se listan algunos de los posibles trabajos futuros:

- Incorporar al sistema una solución al problema que se puede presentar si el robot no detecta etiquetas y se encuentra perdido o se encuentre en un mínimo local. Esta tesis supone que el robot cuando explore el entorno encontrará una etiqueta o señal. En el caso de que ésto suceda, este sistema de navegación se debe complementar con otro tipo de sistemas de navegación o con otro tipo de señales, por ejemplo: señales visuales pertenecientes al entorno, aprendizaje del camino que el robot haya realizado previamente antes de perderse o pedir ayuda a personas que se encuentre en el entorno y seguir las indicaciones que estas le proporcionen.
- Enriquecer los diálogos para añadir mas naturalidad al sistema. Por ejemplo, que el robot pueda entender frases direccionales coloquiales como *“por ahí”*, o poderle dar indicaciones al robot usando una descripción más compleja del entorno, por ejemplo *“La habitación que estás buscando está dos habitaciones después de la sala de fotocopias”*.

- Probar el sistema de navegación en otros entornos para comprobar su rendimiento y robustez, como en hospitales, centros comerciales, museos, etc. Donde el sistema de navegación realizado en esta tesis puede ser útil.
- Desarrollar otro caso posible de automatización del proceso de señalización basado únicamente en exploración. Sería interesante que el robot en base a los movimientos que realice entre etiqueta y etiqueta pueda deducir la información a escribir en ésta, como se ha planteado en el apartado 5.2.2.
- En esta tesis se han presentado los conceptos Lugar, Conexión y Acción. Una posible línea de trabajo en base a estos conceptos sería la Navegación Semántica, donde cada uno de estos elementos tenga un significado semántico para el robot. Por ejemplo, que al indicarle que vaya al “*Laboratorio de Robótica*” sepa que es “*su*” laboratorio, o al decirle que vaya al “*Despacho 1.3.B.15*”, lo relacione con que es “*el despacho de Ana*”.
- Otra posible línea de investigación, particularmente en el campo de la robótica social y en la interacción humano-robot, es el estudio de la generación de emociones por consecuencia de la navegación: estudiar sentimientos como la ansiedad, inseguridad o satisfacción, derivados del hecho que el robot esté cerca o lejos de su destino, tal y como hacemos los humanos cuando, por ejemplo, no recordamos donde hemos dejado el coche aparcado en el estacionamiento o no encontramos un destino específico.

### 8.3. Comentarios Finales

Esta tesis es una pequeña base para los posibles avances que se pueden realizar para la navegación tanto para los robots como para las personas. El uso de la tecnología de RFID para leer señales en el entorno no solo es aplicable a la navegación de robots, sino también para complementar la información a una persona que posea un móvil con tecnología NFC (Near Field Communication<sup>1</sup>) [Tuikka et al., 2011]. Tal vez en un futuro, al igual que hoy encontramos señales visibles en la calle, existan “*ciudades inteligentes*” (“*smart cities*” [Caragliu et al., 2009]) que tengan calles con “*señales inteligentes*”, capaces de comunicarse con las personas a través de dispositivos móviles, enviar información a automóviles y por qué no, a robots que convivan en el entorno.

---

<sup>1</sup> NFC es una extensión del estándar ISO 14443 (RFID)

## Apéndice



# A. HERRAMIENTA PARA LA CONSTRUCCIÓN DE MAPAS GEOMÉTRICOS

La localización es uno de los componentes fundamentales de la navegación de robots móviles, el robot debe determinar su posición en relación a un mapa del entorno, para esto el modelo del entorno o mapa debe ser conocido a priori por el robot. El problema de la localización ha sido ampliamente estudiado, una de las técnicas mas usadas para resolverlo es la aproximación probabilística para estimar la posición del robot, específicamente el uso de *Filtro de Partículas*. Adicionalmente está el problema de generar mapas, se debe tomar en cuenta factores que son importantes para obtener una representación precisa como las dimensiones del entorno, ruido en los sensores y actuadores, percepción ambigua y número de ciclos en el recorrido del robot a través del entorno.

## A.1. SLAM: Simultaneous Localization and Mapping

El problema del SLAM surge cuando el robot no tiene acceso a un mapa del entorno y no conoce su posición, por lo que el robot debe hacer un mapa de su entorno mientras simultáneamente se localiza en el mismo [Cheeseman et al., 1997]. El hecho de hacerlo simultáneamente eleva la complejidad del problema, ya que la localización se puede realizar si se tiene un mapa preestablecido del entorno y la construcción del mapa si se conoce la posición del robot.

Segun Sebastian Thrun [Thrun et al., 2005], desde un punto de vista probabilístico, existen dos tipos de SLAM:

1. SLAM online: Estima la posición y el mapa en un tiempo  $t$  dadas las medidas anteriores y las entradas de control.

$$p(x_t, m \mid z_{1:t}, u_{1:t})$$

Donde  $x_t$  es la posición del robot en un tiempo  $t$ ,  $m$  es el mapa y  $z_{1:t}$  y  $u_{1:t}$  son las medidas tomadas y las entradas de control respectivamente. Es llamado online, ya que estima solo las variables que persisten a través del tiempo.

2. Full SLAM: Aquí el mapa y la posición son estimados tomando en cuenta el camino completo recorrido por el robot, es decir sobre el camino  $x_{1:t}$  en lugar de tomar en cuenta sólo la posición actual.

$$p(x_{1:t}, m \mid z_{1:t}, u_{1:t})$$

## A.2. PMAP Particles Mapping

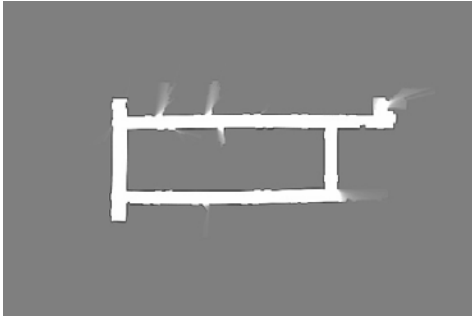
PMAP es una herramienta desarrollada por Andrew Howard en la Universidad del Sur de California [Howard, 2006], es un software con diferentes utilidades para hacer SLAM en entornos 2D. Está compuesto por 4 elementos principales:

1. Una biblioteca de software para corregir errores de estimación de odometría denominada ***lodo*** *laser-stabilized odometry*, usa SLAM para corregir los errores de orientación del robot, trabaja con 3 tipos de estructuras de datos:
  - Lectura actual del laser.
  - Un mapa local, formado por recientes lecturas del laser.
  - Función del error en coordenadas polares, es un grid obtenido a partir del mapa local.
2. Un módulo para aplicar filtros de partículas en posibles mapas del entorno, denominado ***pmap*** o *particle-filter-based mapping*, el resultado final es un mapa que aunque topológicamente correcto posee ciertos desalineamientos. Para mejores resultados es necesario que el mapa sea refinado usando la biblioteca *rmap*.
3. ***rmap***, es un módulo que se usa para refinar o alinear topológicamente los errores existentes en el mapa y así conseguir una aproximación óptima del entorno.
4. La biblioteca ***omap*** *occupancy grid mapping*, que convierte un set de lecturas del laser a un mapa representado como un grid de ocupación.

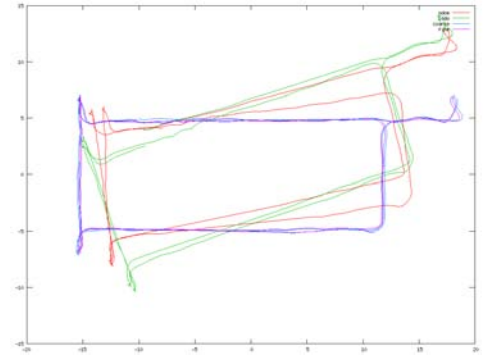
Para obtener buenos resultados los cuatro elementos anteriores deben trabajar en conjunto para obtener un mapa de mayor precisión, las figuras A.1 y A.2 muestran un mapa obtenido con esta técnica y un gráfico comparativo de la localización del robot.

Con esta herramienta se ha realizado un experimento para hacer el mapa correspondiente al Laboratorio 1.3C14 y el pasillo de la 3ra planta zona C del Edificio Betancourt UC3M, usando el joystick movimos el robot a la vez que se tomaban las medidas de odometría y laser. El resultado puede verse en la figura A.3.





**Fig. A.1:** Mapa realizado con pmap en la USC



**Fig. A.2:** Gráfico del error absoluto: odometría vs pmap



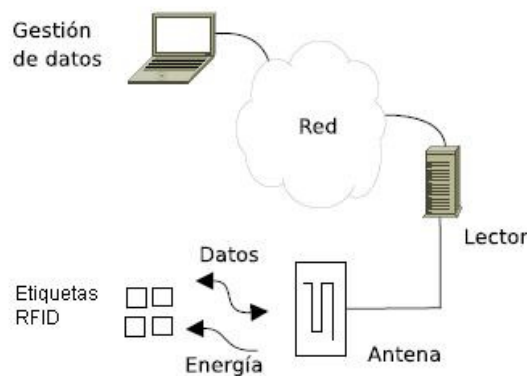
**Fig. A.3:** Laboratorio 1.3C14 y pasillo 3ra planta Edif. Betancourt UC3M usando pmap



## B. SISTEMAS DE IDENTIFICACIÓN POR RADIO FRECUENCIA

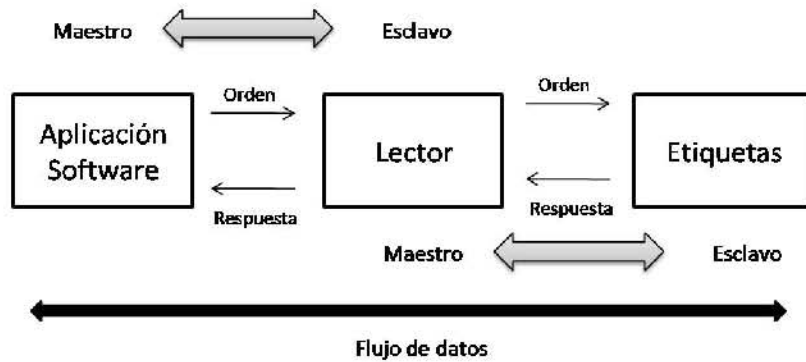
La identificación por Radio Frecuencia o RFID (Radio Frequency IDentification), utiliza ondas electromagnéticas o electrostáticas para la transmisión de la señal que contiene la información, por lo que nos permite identificar objetos a distancia. Una etiqueta RFID es capaz de guardar y enviar información a un lector RFID, ya que incluye un circuito integrado con una memoria para almacenar información y una antena. En el circuito integrado se encuentra la lógica para responder al lector, la antena emite y recibe las ondas de radio para intercambio de información [RFI, 2006, Shepard, 2005].

Un sistema RFID está compuesto por 3 elementos básicos: Lector, antena y etiqueta (Figura B.1).



*Fig. B.1: Elementos de un sistema RFID*

1. **Lector:** Un lector (o interrogador), es un dispositivo capaz de leer y/o escribir datos en las etiquetas. El lector produce una señal que va hacia una antena que radia la misma señal en el espacio, a una frecuencia determinada, para que otros elementos la escuchen. El lector posteriormente escucha la respuesta de las etiquetas. Transmite y recibe ondas que transforma en información digital. Las operaciones de lectura y escritura siguen el principio *maestro-esclavo*, esto significa que el lector es activado sólo cuando recibe la orden desde una aplicación software, igualmente, las etiquetas solo responden cuando el lector inicia comunicación con éstas [Finkenzeller, 2004] (Figura B.2).



**Fig. B.2:** Principio Maestro-Esclavo entre la aplicación software, lector y etiquetas.

Existen diferentes tipos de lectores: simples (un solo estándar y frecuencia), multiregionales y multifrecuencias; también pueden ser fijos (ej. para leer elementos de una cinta transportadora o control de acceso) o móviles (para identificar elementos que pueden estar en cualquier lugar) [Huidobro, 2004].

2. **Antena:** Es el dispositivo que permite radiar la señal del lector que activa las etiquetas que se encuentran en su rango de lectura y además recibe las ondas de radio de las etiquetas. La energía para activar el chip la toma del campo de Radio Frecuencia (campo electromagnético para HF y campo eléctrico en UHF).

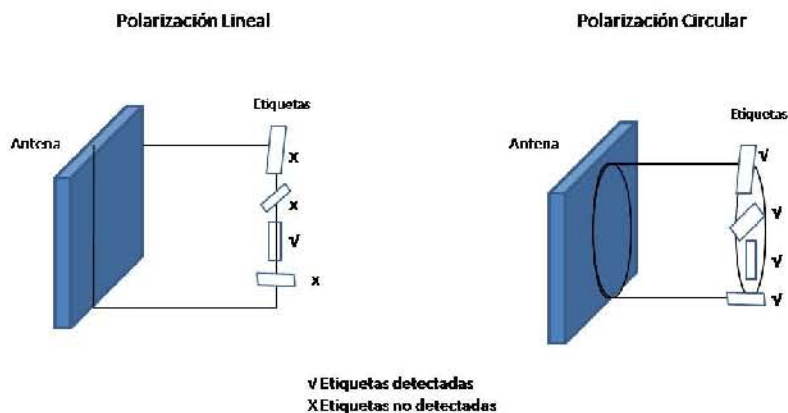
Es posible encontrar lectores con antenas integrados en un mismo elemento. También varias antenas pueden ser gestionadas por un mismo lector/grabador.

La polarización de la antena dependerá de la aplicación (Figura B.3):

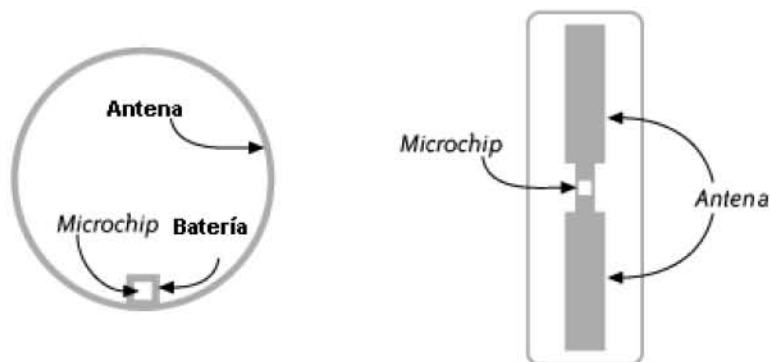
- **Polarización Lineal:** Para aplicaciones donde la orientación de la antena debe ser fija y las etiquetas estar alineadas para su lectura o escritura.
  - **Polarización Circular:** Es usada en lectores móviles, éstas antenas son usadas en situaciones donde la orientación de las etiquetas no es controlada.
3. **Etiquetas:** Una etiqueta (o transponder) RFID es capaz de guardar y enviar información a un lector RFID. Incluye un circuito integrado con una memoria para almacenar información y una antena. En el circuito integrado o chip se encuentra la lógica para responder al lector (Figura B.4) [Bhatt and Glover, 2006]. La antena emite y recibe las ondas de radio para intercambio de información con el lector.

Las etiquetas, según el tipo de alimentación, se clasifican en:

- **Activas:** No necesitan la energía de la antena para activarse ni para enviar datos, ya que poseen una batería propia. Permiten mayor alcance de comunicación (hasta 30 metros aprox.), poseen mayor capacidad de almacenamiento de datos y algunas pueden incluir sensores adicionales como



*Fig. B.3: Tipos de Polarización de las Antenas RFID*



*Fig. B.4: Etiquetas RFID*

sensores de temperatura. Las principales desventajas de este tipo de tarjetas es su límite de vida útil (depende de las baterías), su tamaño es mucho mayor en comparación con las tarjetas pasivas y su costo de mantenimiento es mayor [TN, 2006].

- **Semi-Activas:** Este tipo de tarjetas posee una batería propia para activar su circuitería para recibir datos. Para enviar información, toma la energía de las ondas de radio emitidas por la antena. Son más costosas que las pasivas pero proporcionan mayor rango de comunicación (entre 10 y 15 metros aprox.).
- **Pasivas:** Las etiquetas pasivas no poseen una fuente de alimentación incorporada. Para activarse y transmitir datos, usan el campo electromagnético generado por la antena del lector. Las ventajas de este tipo de etiquetas son su bajo coste de adquisición y de mantenimiento, su tamaño es más pequeño que las activas y semi-activas. La principal desventaja es la corta distancia para el rango de lectura (entre 3 cm y 9 metros aprox.).

## B.1. Ventajas y Limitaciones de los sistemas RFID

Como toda tecnología en desarrollo, RFID posee diversas ventajas y ciertas limitaciones, en [Corrales, 2008] se destacan:

### 1. Ventajas

- a) Resistencia de las etiquetas a fuertes condiciones ambientales, soporta la humedad, suciedad y los cambios drásticos de temperatura.
- b) Rapidez de respuesta (menos de 100ms), un lector puede leer cientos de etiqueta en muy corto tiempo.
- c) Las etiquetas RFID pueden ser leídas sin necesidad de contacto físico con el lector, o de una lectura visible y directa (como los códigos de barras).
- d) Las etiquetas pueden escribirse un gran número de veces.
- e) El rango de lectura es variable, desde pocos centímetros hasta mas de 100 metros (en UHF).
- f) Mayor capacidad de almacenamiento que los códigos de barras (etiquetas pasivas 64KB, etiquetas activas mas de 8MB).
- g) La tecnología RFID permite leer múltiples etiquetas electrónicas simultáneamente. Los códigos de barras, por lo contrario, tienen que ser leídos secuencialmente.
- h) Se puede programar el lector para que busque un objeto específico.
- i) Es posible codificar toda o parte de la información almacenada.

### 2. Limitaciones

- a) La eficiencia de la lectura es proporcional al tamaño de la antena y de las etiquetas. Etiquetas pequeñas necesitan antenas grandes y viceversa.
- b) En ciertas condiciones del entorno no puede trabajar eficientemente. Las ondas de radio se atenúan al pasar por ciertos medios (en particular el agua) y se reflejan al chocar con metales, lo cual reduce notablemente el radio de lectura
- c) El tiempo de vida de las etiquetas activas depende de la duración de sus baterías.

## C. HABILIDADES BÁSICAS PARA NAVEGACIÓN

A continuación se presentan las diferentes habilidades de navegación diseñadas, desarrolladas e implementadas en el robot Maggie, para la elaboración de esta tesis.

### C.1. Habilidad de Localización Geométrica

La *Localización Geométrica* la usamos para localizar el robot en entornos no estructurados donde un robot requiere una localización precisa, para esto es necesario contar con un mapa geométrico del entorno. Para esta tesis, los mapas se han realizado usando la herramienta `pmap`, explicada previamente en el apéndice A.

El algoritmo de localización que utilizamos es el AMCL (Adaptive Monte Carlo Localization) [Fox et al., 1999], es un algoritmo de localización probabilístico que calcula la localización de un robot móvil en entornos 2D. Utiliza un filtro de partículas<sup>1</sup> para calcular la posición de un robot con respecto a un mapa geométrico conocido. La habilidad de Localización Geométrica que hemos desarrollado usa el algoritmo implementado en el driver de Player<sup>2</sup> realizado por Andrew Howard.

La habilidad que se ha desarrollado usa un mapa geométrico y los datos de la odometría y el láser como entrada del sistema de localización, devuelve la posición geométrica del robot con respecto al mapa geométrico proporcionado.

El funcionamiento de la habilidad es el siguiente:

- Inicialmente la habilidad se suscribe a dos eventos usados para activar y bloquear la habilidad.
- Cuando la habilidad recibe el evento para activarse, inicia la función `proceso()`. Esta función solicita al algoritmo de localización el número de hipótesis que representan los posibles puntos donde se cree que el robot se encuentra ubicado geométricamente. Mientras menor sea el número de hipótesis, el robot estará localizado con mayor precisión. Si el número de hipótesis obtenido es menor o igual a un número de  $N$  de hipótesis mínimas, entonces decimos que el robot está localizado. De las  $N$  hipótesis mínimas se escoge la que tenga mayor peso

---

<sup>1</sup> El filtro de partículas es un método empleado para estimar el estado de un sistema que cambia a lo largo del tiempo. Se compone de un conjunto de muestras (partículas) y unos valores o pesos, asociados a cada una de esas muestras. Las partículas son los posibles estados del proceso.

<sup>2</sup> Player & Stage [googl.github.io/NICRR](http://googl.github.io/NICRR)

o mayor probabilidad y se actualizan los datos de odometría del robot con los datos de la posición estimada obtenida.

En el algoritmo C.1, se muestra el contenido de la función `proceso()` de la habilidad.

**Algoritmo C.1:** *Función `proceso()` de la Habilidad de Localización Geométrica*

```
void CSkill_Geometric_Localize::proceso(){
    this->robot->Read();
    nh = localizador->GetNumHypothesis();

    if(nh<=HIPOTESIS_MINIMAS){
        for(i=0; i<nh; i++){
            this->datos_hipotesis =localizador->GetHypothesis(i);

            // Busco la hipótesis con mayor peso
            if((this->datos_hipotesis.alpha)>peso_mayor){
                iw = i;
                peso_mayor = this->datos_hipotesis.alpha;
            }
        }

        std::cout<<"Encontre "<<nh<<" hipótesis minimas"
                <<"...seleccione la "<<iw<<"\n";
        this->datos_hipotesis = localizador->GetHypothesis(iw);
        data->x = (this->datos_hipotesis.mean.px)*1000; //paso a mm
        data->y = (this->datos_hipotesis.mean.py)*1000;
        data->theta = (this->datos_hipotesis.mean.pa); //en radianes

        //Actualiza la posición estimada en la odometría del robot
        this->ruedas->setEstimatedPose(data);

        if(LOCALIZADO==0){
            LOCALIZADO = 1;
        }
    }
    peso_mayor =0;
}
```

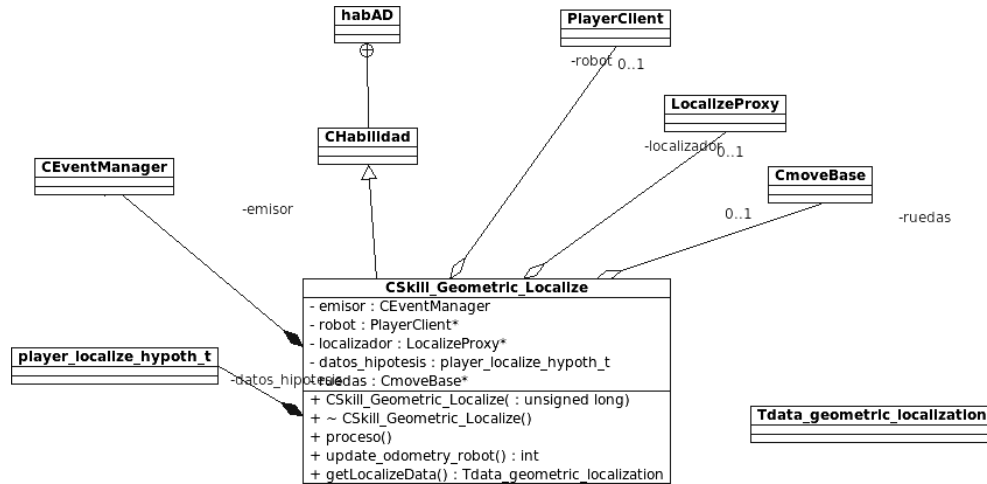
La figura C.1 muestra el diagrama de clases de la Habilidad de Localización Geométrica. Esta habilidad hereda de la clase base `CHabilidad` de la arquitectura AD. Instancia un cliente de `Player` para usar el algoritmo AMCL de Localización usando la interfaz `LocalizeProxy`<sup>3</sup>. También usa la clase `CmoveBase` que es un objeto para acceder la base móvil del robot y así obtener/actualizar los datos de la odometría.

### C.1.1. Cómo Localizar el Robot Geométricamente en el Robot Maggie

**Requerimientos:** Para localizar el robot geométricamente es necesario tener instalado el Player <http://playerstage.sourceforge.net>. Debe estar el servidor del

<sup>3</sup> Localize – Interfaz specifications (Player) [googl/Ujja1](http://googl/Ujja1)





**Fig. C.1:** Diagrama de clases de la habilidad de Localización Geométrica.

Láser y el servidor del movimiento de la Base en ejecución.

### Pasos a seguir:

- Ejecutar el comando `player robot_maggie.cfg`

Donde `robot_maggie.cfg` es un fichero de configuración donde se indica la fuente de los datos de odometría y láser del robot y el mapa geométrico donde se localizará el robot<sup>4</sup>.

**Algoritmo C.2:** Ejemplo del fichero de configuración de Player para la localización.

```

driver(
  name "robot_maggie"
  provides ["laser:0"]
  rate 5
)

driver(
  name "robot_maggie"
  provides ["position2d:0"]
  rate 5
)

driver(
  name "mapfile"
  provides ["map:0"]
  filename "mapa_3cm_centrado_fit.png"
  resolution 0.03
)

```

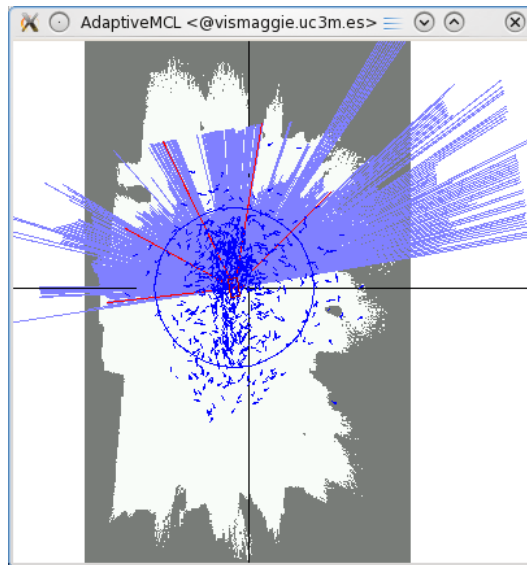
<sup>4</sup> Para el sistema de navegación de esta tesis, el programador no se debe preocupar por ejecutar este comando, ya que dependiendo de lo que lea el robot en la señal, se ejecutará una hebra que lo ejecutará con el mapa geométrico que indique la señal.

```

driver(
  name "amcl"
  provides ["localize:0"]
  requires ["odometry::position2d:0" "laser:0" "laser::map:0"]
  enable_gui 1
  pf_max_samples 50000
  init_pose [-1.8953 0.000954025 10]
  #init_pose recibe el angulo en grados
  init_pose_var [1 1 0]
  alwayson 1
)

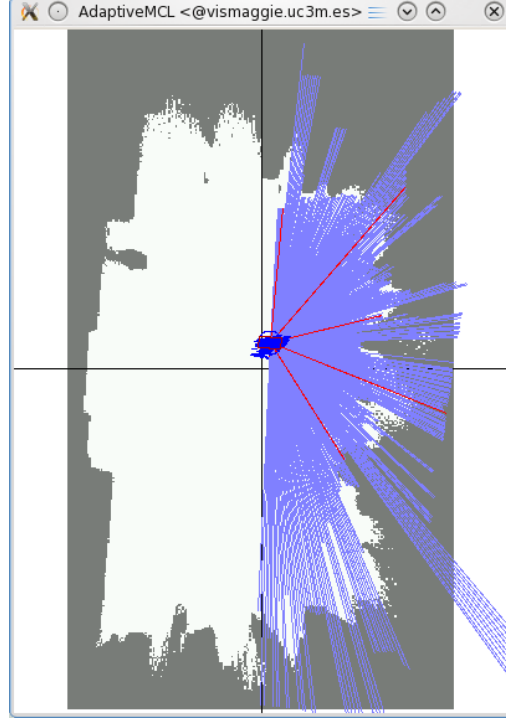
```

- Aparecerá una pantalla el mapa que se haya asignado en el fichero de configuración, las posiciones estimadas y el barrido del láser (Fig. C.2).



**Fig. C.2:** Imagen del mapa del Laboratorio de Robótica con el robot no localizado.

- Activar la habilidad `CSkill_Geometric_Localize`.
- Mover el robot para empezar a localizarlo, en la imagen se verá como se juntan las partículas (o hipótesis) hasta que el robot se localice (Fig. C.3).



**Fig. C.3:** Imagen del mapa del Laboratorio de Robótica con el robot localizado.

## C.2. Habilidad “Ir a un punto”

Esta habilidad mueve al robot desde un punto geométrico de inicio a otro punto geométrico destino. Para lograr este objetivo se debe diseñar un controlador de posición del robot móvil, tomando en cuenta que en la base móvil donde se implementará es no holonómica. Lo que se pretende controlar es la posición del robot (variable controlada) usando la velocidad angular y la velocidad lineal (variables manipuladas).

### C.2.1. Control Cinemático de la Posición

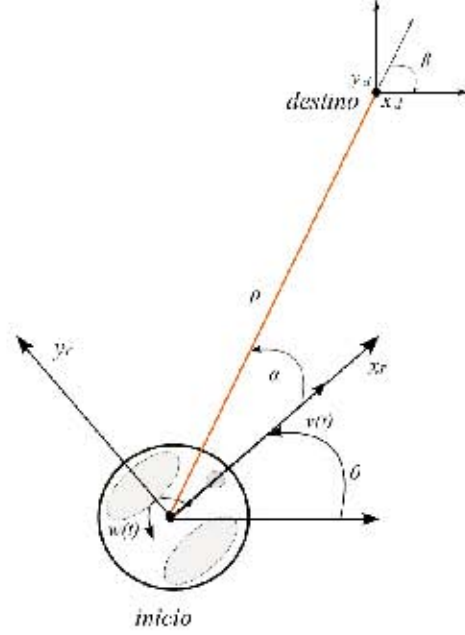
La posición geométrica de un robot con base diferencial esta representado por un vector con las variables  $(x, y, \theta)$  que representan la posición del robot en el plano cartesiano y su orientación. El modelo cinemático está representado por la siguiente ecuación:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (\text{C.1})$$

Donde  $\dot{x}$  e  $\dot{y}$  corresponden a la velocidad lineal en la dirección de la posición destino  $x_d$  e  $y_d$  respectivamente, del sistema de coordenadas cartesianas.

En la figura C.4 se muestra el gráfico del modelo cinemático de la posición del robot. La variable  $\alpha$  corresponde al ángulo entre la posición inicial del robot y el vector que conecta el centro del robot a la posición destino. La distancia mínima

entre los puntos de origen y destino está dada por la variable  $\rho$  (Ecuaciones C.2 y C.3) [Siegwart and Nourbakhsh, 2004].



**Fig. C.4:** Modelo cinemático del robot para la habilidad *Ir a un Punto*

$$\rho = \sqrt{\Delta x^2 + \Delta y^2} \quad (C.2)$$

$$\alpha = \theta - \text{atan2}(\Delta y, \Delta x) \quad (C.3)$$

El controlador que se diseñó para la habilidad *Ir a un punto*, es un sistema en lazo cerrado. El controlador se alimenta de la señal de error de la posición, que es la diferencia entre la señal de entrada o posición actual y la señal de realimentación.

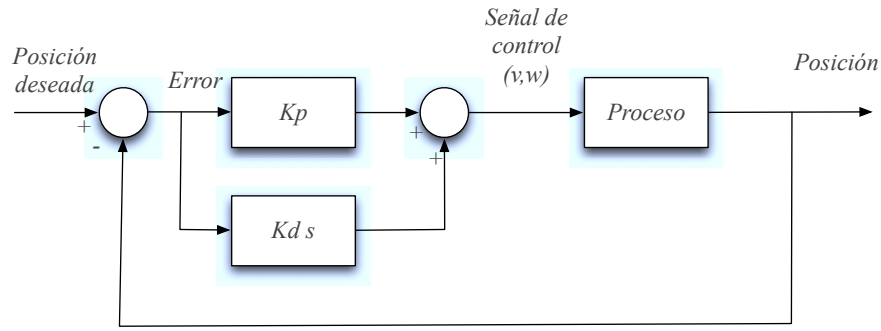
El controlador diseñado es un controlador PD (Proporcional-Derivativo), La parte derivativa del controlador mantiene el error de posición al mínimo corrigiéndolo proporcionalmente con la misma velocidad que se produce.

Para la habilidad se realizaron dos controladores PD, uno para el control de la velocidad lineal y angular y el segundo controlador hace la corrección final de la orientación del robot al llegar al punto destino.

El error de posición escogido es de 5 cm, y el error angular es de 6 grados.

A continuación se muestran las diferentes etapas del controlador implementadas en el proceso principal de la habilidad *Ir a un punto* (CSkillGotoPoint).

1. **Adquisición de datos:** Inicialmente se comprueba que el robot tiene asignado un *punto destino*. Posteriormente se obtiene la posición geométrica del robot (datos de la odometría).



**Fig. C.5:** Controlador Proporcional-Derivativo para la habilidad “Ir a un punto”

**Algoritmo C.3:** Adquisición de datos

```

86  this->d = ruedas->getEstimatedPose();
87  angulo = d->theta;
88  angulo = set_range_angle(angulo);
89  this->actual.setCoord((d->x)/1000,(d->y)/1000, angulo ); //En m y rad

```

La línea 88 del código mostrado en el algoritmo C.3, llama a la función `set_range_angle()` que convierte el rango del ángulo en radianes de  $-\pi$  a  $\pi$ .

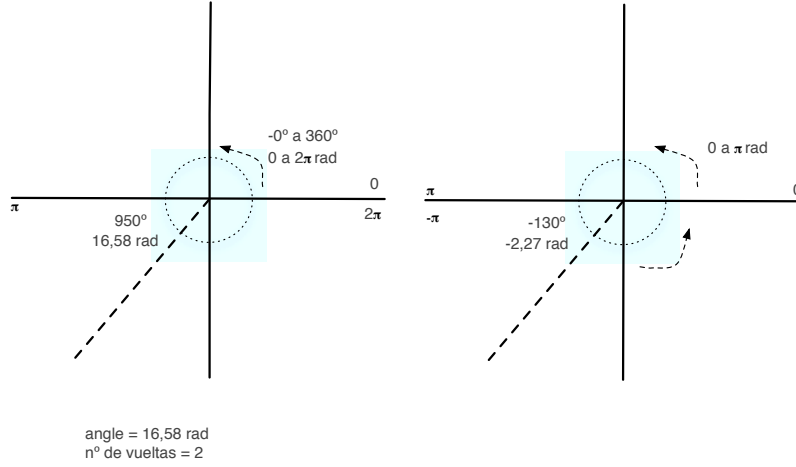
**Algoritmo C.4:** Función `set_range_angle`

```

1  float set_range_angle(float angulo){
2      float a, pi;
3      int vueltas;
4      pi = M_PI;
5      a = fmod(angulo,pi);
6      vueltas = (int)abs(angulo/(2*M_PI));
7
8      if(angulo>=0){
9          if (angulo>2*M_PI)
10             angulo = angulo-vueltas*2*M_PI;
11
12             if(angulo>M_PI)
13                 angulo = a-M_PI;
14         }
15     else{
16         if (angulo<-2*M_PI)
17             angulo = angulo+vueltas*2*M_PI;
18
19             if(angulo<-1*M_PI)
20                 angulo = M_PI+a;
21         }
22     return angulo;
23 }

```

El algoritmo C.4, muestra el cambio de rango del ángulo que se obtiene de la odometría de  $-\pi$  a  $\pi$ . En la figura C.6 se muestra el cambio de rango gráficamente.



**Fig. C.6:** Cambio de rango del ángulo de  $-\pi$  a  $\pi$ .

## 2. Controlador 1: Control de la velocidad lineal y angular

Sean  $x_d$ ,  $y_d$  y  $\theta_d$  las coordenadas del punto hacia donde debe ir el robot (destino),  $x_a$ ,  $y_a$  y  $\theta_a$ , las coordenadas de la posición actual del robot,  $emin_\alpha$  y  $emin_\rho$  los errores mínimos angular y de posición que se desean alcanzar. Entonces el controlador esta dado por las siguientes ecuaciones:

$$\Delta x = x_d - x_a$$

$$\Delta y = y_d - y_a$$

$$\lambda = \text{atan2}(\Delta y, \Delta x)$$

$$\alpha = \lambda - \theta_a$$

$$\beta = \theta_a - \lambda$$

Inicialmente, el error angular  $e_\alpha$ , está dado por:

$$e_\alpha = \alpha$$

y el error con respecto a la distancia  $e_\rho$ , está dado por:

$$e_\rho = \rho$$

La velocidad angular entonces, estará dada por la siguiente ecuación:

$$\omega = K_\alpha * \alpha + KD_\alpha * \left( \frac{\alpha - e_\alpha}{T} \right)$$

donde  $K_\alpha$  es la constante proporcional del controlador,  $KD_\alpha$  es la constante derivativa y  $T$  es el período de muestreo.

El control de la velocidad lineal está dado por:

$$v = K_\rho * \rho + KD_\rho * \left( \frac{\rho - e_\rho}{T} \right)$$

donde  $K_\rho$  es la constante proporcional del controlador,  $KD_\rho$  es la constante derivativa y  $T$  es el período de muestreo.

Cuando el error  $e_\rho$  sea menor o igual al error mínimo  $emin_\rho$ , decimos que el robot ha alcanzado su posición geométrica. Una vez alcanzada la posición se procede a corregir el ángulo de orientación final.

### 3. Controlador 2: Control de la posición angular final

El objetivo de este controlador es corregir la orientación final del robot. El controlador está dado por las siguientes ecuaciones:

$$\Delta\theta = \theta_d - \theta_a$$

La velocidad angular será controlada a través de la siguiente ecuación de un controlador PD:

$$\omega = K_\alpha * \alpha + K D_\alpha * \left(\frac{\alpha - e_\alpha}{T}\right)$$

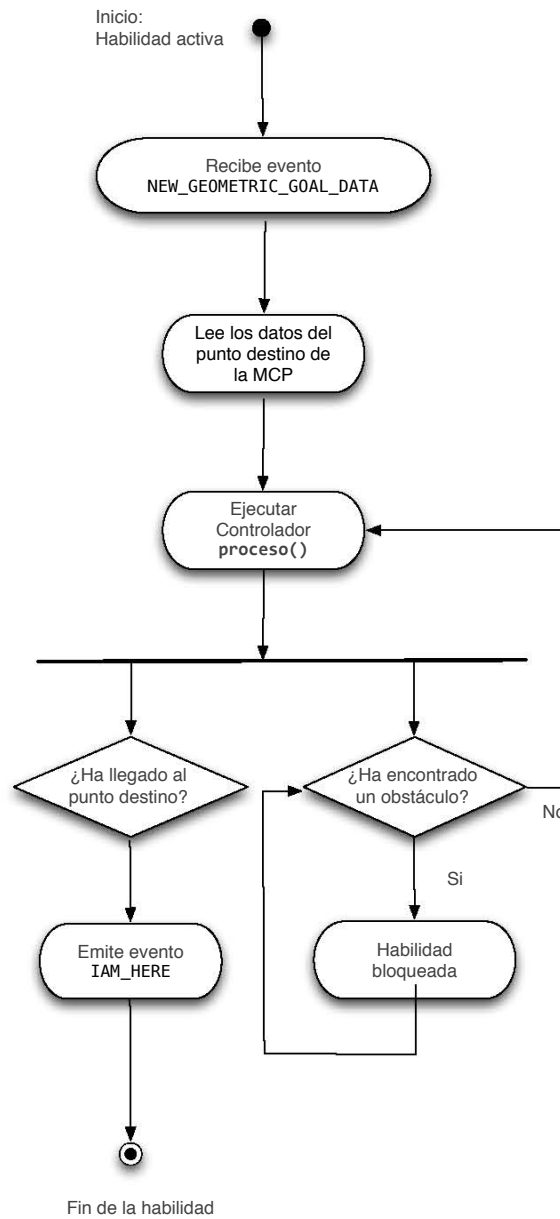
Cuando  $\Delta\theta$  sea menor que el error mínimo angular  $emin_\alpha$ , entonces decimos que el robot ha alcanzado la orientación correcta.

## C.2.2. Comportamiento de la Habilidad “Ir a un punto”

Al igual que el resto de habilidades que forman parte de la arquitectura AD, esta habilidad usa el Manejador de Eventos para recibir y enviar notificaciones, así como también usa la Memoria a Corto Plazo para la comunicación de datos. La figura C.7 muestra el diagrama de actividades de la habilidad.

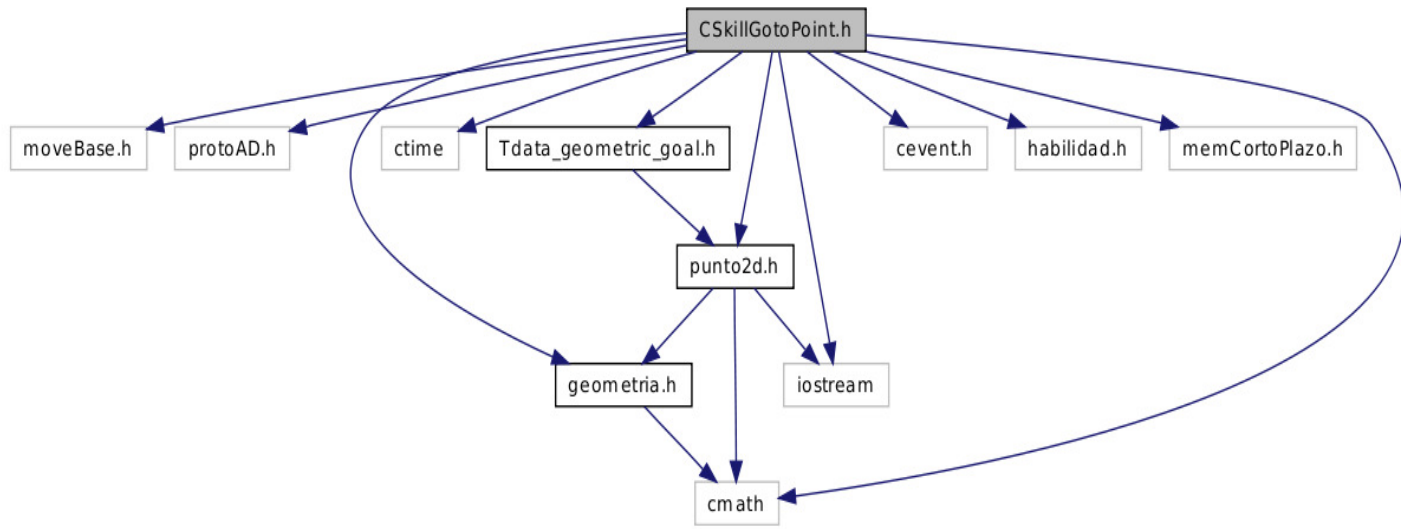
Cuando se activa la habilidad, ésta espera a recibir el evento que notifica a la habilidad que hay una meta geométrica asignada (o coordenada destino). La habilidad lee los datos de la Memoria a Corto Plazo y ejecuta el proceso principal de la habilidad donde se ha implementado el controlador. Cuando el robot llega al punto geométrico deseado la habilidad emite el evento `IAM_HERE` para notificar al sistema o a otras habilidades que se ha cumplido el objetivo satisfactoriamente. Si el robot al ejecutar la habilidad encuentra un obstáculo, la habilidad se bloquea y espera a que otra habilidad (por ejemplo, la habilidad de “Evitar obstáculos”), maneje el imprevisto y cuando ya no haya un obstáculo la habilidad continuará ejecutando el proceso con la meta que se le ha asignado previamente.

A continuación, en la figura C.8, se muestra el grafo de dependencias de la habilidad “Ir a un punto”, desarrollada en C++.



**Fig. C.7:** Diagrama de actividades de la habilidad Ir a un Punto.





**Fig. C.8:** Grafo de dependencias de la habilidad *Ir a un Punto*.

La habilidad `CSkillGotoPoint`, hereda de la clase base `habilidad` de la arquitectura AD integrada en el robot Maggie, donde se ha implementado esta tesis. Como se puede ver la habilidad usa el Manejador de eventos y la Memoria a Corto Plazo del sistema para comunicarse con otras habilidades. También usa las funciones que le proporciona la clase que maneja la base del robot `moveBase.h`, para acceder a los datos de odometría y para asignar la velocidad lineal y angular. En `Tdata_geometric_goal.h` se define el tipo de dato que se lee de la Memoria a Corto Plazo.

Adicionalmente, para facilitar el acceso a las funciones de la habilidad “Ir a un punto”, se desarrolló una clase interfaz adicional llamada `CApiSkillGotoPoint`, con dos funciones principales:

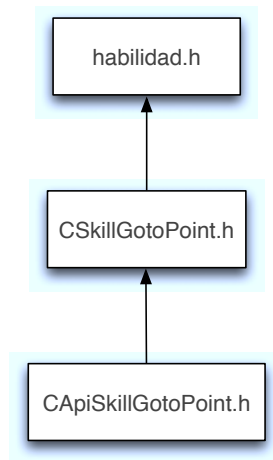
- `intCApiSkillGotoPoint :: asignarCoordenadasDestinoAbsoluto(CPunto2d * p, int iterador)`

donde  $p$  es un objeto `CPunto2d` para asignar la coordenada geométrica del punto destino y el parámetro `iterador` define el tipo de meta, si es parcial  $>2$ , si es final `iterador = 2`.

Se ha agregado la opción de determinar si la meta es parcial ya que si es necesario dirigir el robot por una serie de puntos geométricos consecutivos, éste pueda hacer una navegación continua entre los puntos (sin detenerse cada vez que llegue a uno de los puntos), si se encuentra en este caso, los errores mínimos de posición y angular serán mas amplios.

- `CApiSkillGotoPoint :: obtenerCoordenadasPosicionActual(CPunto2d * pa):`  
Obtiene la coordenada de la posición actual. En metros y radianes.

La figura C.9, muestra el diagrama de herencia de la habilidad *Ir a un Punto* con la clase `CApiSkillGotoPoint` incluida.



**Fig. C.9:** Diagrama de herencia de la habilidad *Ir a un Punto*.

### C.3. Habilidad Seguir Contorno

Esta habilidad hace que el robot siga el contorno de una pared y se mantenga con respecto a ésta a una determinada distancia o rango de distancias.

Para la habilidad “*Seguir contorno*” se diseñó un controlador proporcional (P). El objetivo del controlador es tratar de mantener el robot a una distancia mínima aproximada (DLATERAL\_MIN) durante la ejecución de la habilidad. El sensor que se utilizó fue el telémetro láser SICK LMS 200 incorporado en el robot Maggie, ya que proporciona medidas exactas de la distancia y, por su rango de lectura (360 medidas en 180 grados), proporciona mayor precisión al controlador.

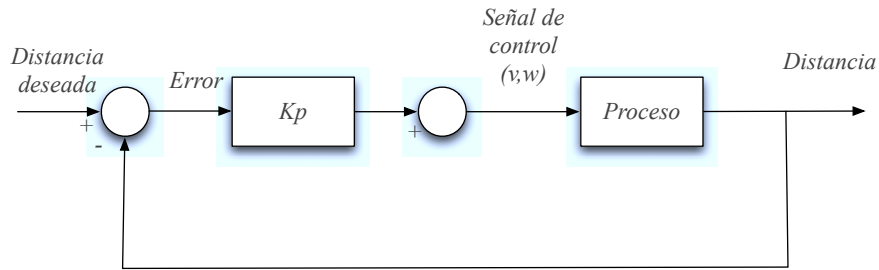
#### C.3.1. Control Proporcional para la habilidad *Seguir contorno*

El controlador diseñado es un sistema en lazo cerrado. El sistema responde en proporción al error de salida del sistema. Un controlador P produce una salida  $y$  proporcional a una entrada  $i$  (Ecuación C.4).

$$y = K_p e \quad (C.4)$$

Donde  $K_p$  es una constante proporcional y  $e$  el error o la diferencia entre la salida del sistema y la señal deseada ( $Sd$ ).

$$e = Sd - i \quad (C.5)$$

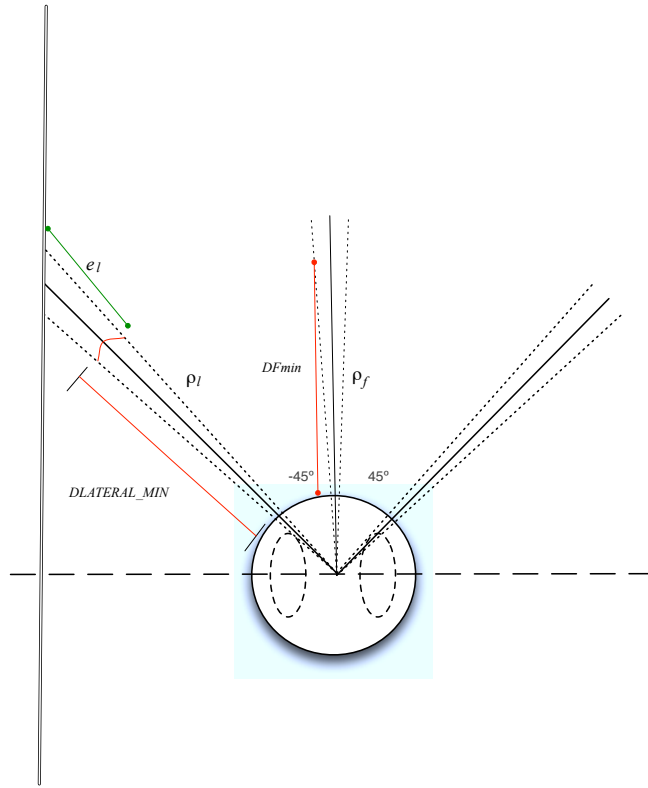


**Fig. C.10:** Controlador Proporcional para la habilidad “Seguir Contorno”

En la figura C.11 se muestra el modelo con las variables utilizadas para la Habilidad “Seguir Contorno”.

A continuación se explican las diferentes etapas del controlador implementadas en el proceso principal de la habilidad *Seguir contorno* (CSkillWallFollowing).

1. Se inicializan las constantes del sistema. La distancia lateral mínima la asignamos en 90 cm y la distancia frontal en 100 cm. Se tomarán en cuenta 12 lecturas del láser como campo de visión lateral (FOV), correspondientes a 6 grados aproximadamente.



**Fig. C.11:** Modelo para la habilidad “Seguir Contorno”

2. En la función *proceso* de la habilidad, se ejecutará el controlador para mantener la distancia con la pared y para hacer girar el robot en caso que se encuentre con una pared al frente.
3. El sistema recibe como entrada el lado en que el robot debe realizar el seguimiento, bien sea por la izquierda o por la derecha. El robot tomará las medidas del láser a  $45^\circ$  ( $\pm 3^\circ$ ) o a  $135^\circ$  ( $\pm 3^\circ$ ) (Fig. C.11).
4. Se escoge el valor mínimo del rango de medidas del lado que se haya determinado seguir.
5. **Controlador:**

Sea  $\rho_l$ , la distancia lateral que mide el láser,  $DLATERAL\_MIN$ , la distancia a la que el robot debe estar separado de la pared para hacer el seguimiento y  $e_l$  el error entre la distancia deseada y la medida por el láser.

El error lateral está dado por la siguiente ecuación:

$$e_l = \rho_l - DLATERAL\_MIN$$

También se considera el error con respecto a una distancia frontal mínima para que el robot realice el giro necesario para continuar con el seguimiento de contorno en el caso que encuentre un obstáculo al frente.

$$e_f = \rho_f - DFmin$$

- Si  $e_f > 0$ , la velocidad lineal del robot  $v$  se mantiene constante, para esta tesis, hemos puesto este valor  $Vconst = 25cm/seg$ .
- Si  $e_f \leq 0$ , entonces la velocidad lineal está dada por:  
 $v = k_p * e_f$   
 Si  $v > Vconst$  entonces  $v = Vconst$ ,  
 Si  $v \leq 0$  entonces  $v = 0$ .
- Con la velocidad angular  $w$ , se controla la distancia lateral,  
 $w = k_p * e_l$   
 La velocidad máxima angular la hemos limitado a una constante  $Wmax = 30^\circ/seg$ .  
 Si  $w > Wmax$  entonces  $w = Wmax$ ,  
 Si  $w < -Wmax$  entonces  $w = -Wmax$ ,
- Si  $\rho_f > DFmin$  y  $\rho_l$  está en el intervalo  $(DLATERAL\_MIN +/\delta)$ , donde  $\delta$  es la precisión  $(+/- 5\text{ cm})$  entonces el robot tendrá las velocidades  $v = Vconst$  y  $w = 0$ .
- En el algoritmo C.5, se muestra el código en C++ del controlador.

**Algoritmo C.5:** Algoritmo de Control para la habilidad Seguir Contorno

```

1 // Recibe como parámetro el lado que el robot debe seguir el contorno
2 void CSkill_WallFollowing::aproximarse_pared(int l){
3     this->v = this->laser.get(); //Obtiene datos del laser
4
5     if (this->lado_rodeo==1)
6         pos_lateral = 180+ (45*2);
7     else
8         pos_lateral = 180- (45*2);
9
10    for (i=pos_lateral-fov/2; i<=pos_lateral+fov/2; i++){
11        dlateral += this->v[i].r;}
12    dlateral/=fov;
13    rho = dlateral;
14    error_lat = rho - DLATERALMIN;
15    a = this->laser.length()/2-fov/2;
16    b = this->laser.length()/2+fov/2;
17    i=this->buscar_mas_cercano(a,b);
18    df = this->v[i].r;
19    error_front = df - DFMIN;
20    if(df>DFMIN){
21        v= VCONST; }
22    else{
23        v = kp*error_front;
24        if(v>VCONST)
25            v= VCONST;

```

```

26     if(v<=0)
27         v = 0; }
28     w = kp*error_lat;
29     if(w>WMAX)
30         w=WMAX;
31     if(w<-WMAX)
32         w=-WMAX;
33
34     if((df>DFMIN)&&(dlateral<(DLATERALMIN+DELTA))&&
35         (dlateral>(DLATERALMIN-DELTA))) {
36         std::cout<<"Debe ir a velocidad lineal constante\n";
37         v = VCONST;
38         w = 0;
39     }
40     if((df<DFMIN)&&(dlateral<(DLATERALMIN+DELTA))&&
41         (dlateral>(DLATERALMIN-DELTA))) {
42         w=30;
43         w*=-1;
44     else {
45         // Si el robot está lejos de la pared y no tiene obstáculo en frente
46         // el robot se aproxima a la pared tomando en cuenta las medidas
47         // más a la izquierda o más a la derecha dependiendo del caso
48         //(Medidas 0°-6° o 172°-180°)
49         if((df>DFMIN)&&(dlateral>DLATERALMIN)) {
50
51             if (this->lado_rodeo==1)
52                 pos_lateral = laser.length()-1-fov;
53             else
54                 pos_lateral = 0;
55             dlateral=0;
56             for (i=pos_lateral; i<=pos_lateral+fov; i++){
57                 dlateral += this->v[i].r;
58             }
59             dlateral/=fov;
60             rho = dlateral;
61             error_lat = rho - DLATERALMIN;
62             w = kp*error_lat;
63             if(w>WMAX)
64                 w=WMAX;
65             if(w<-WMAX)
66                 w=-WMAX;
67         }
68     }
69     //Seguimiento de contorno por la izquierda (1) no se cambia de signo
70     //Seguimiento de contorno por la derecha (0) se cambia de signo
71     if(this->lado_rodeo == 0){
72         w*=-1;
73     }
74     //Asignar velocidad lineal y angular a la base
75     ruedas.setVelocityBase((int)v,(int)w);
76     return;
77 }

```

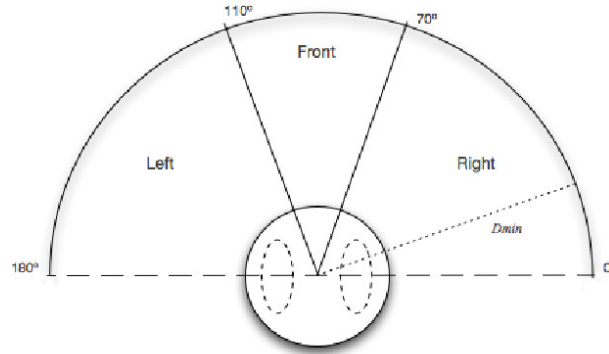
## C.4. Habilidad Evitar Obstáculos

La habilidad de Evitar Obstáculos es el resultado de la combinación de 2 habilidades:

1. Habilidad de Detección de Obstáculos.
2. Habilidad de Seguir Contorno (C.3).

### C.4.1. Habilidad de Detección de Obstáculos

La Habilidad de Detección de Obstáculos, usa el láser para detectar si hay un obstáculo cercano al robot a una distancia  $D_{min}$ . Una solución sencilla consiste en dividir el rango de visión del láser en 3 zonas: Derecha, Frente e Izquierda (Fig. C.12).



**Fig. C.12:** División por zonas del rango de visión del láser.

La habilidad emite el evento **OBSTACLE** si encuentra un obstáculo próximo al robot dentro del rango de distancias menores a una distancia de seguridad mínima establecida  $D_{min}$ . En el evento también se incluye un parámetro  $p$ , que indica la zona (L,F,R) donde se ha detectado. En la tabla Tab.C.1 se observa los valores que puede tomar la variable  $p$  dependiendo de las zonas donde se detecte un obstáculo.

Dependiendo de la zona o zonas donde se detecte el obstáculo y la orientación del punto destino o meta, el robot evitará el obstáculo siguiendo el contorno a su derecha o a su izquierda.

Esto se observa mejor en la habilidad **CSkill\_SafeGoToPoint**, que hace que el robot vaya a un punto destino o meta, evitando los posibles obstáculos que encuentre durante la ejecución de la habilidad “Ir a un punto” (C.2).

<b>L</b>	<b>F</b>	<b>R</b>	<b>p</b>
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

**Tab. C.1:** Posibles valores que indican la zona o zonas donde se han detectado obstáculos.

#### C.4.2. CSkill\_SafeGoToPoint

Esta habilidad controla el comportamiento de 3 habilidades básicas de navegación:

1. Ir a un punto.
2. Detección de obstáculos.
3. Seguir contorno.

El objetivo de esta habilidad es que el robot pueda ir a una meta geométrica determinada de manera segura, es decir, evitando obstáculos, pero sin perder su objetivo.

El funcionamiento de esta habilidad depende de las siguientes variables:

1. La dirección donde se encuentre la meta con respecto al robot.
2. Las zonas próximas al robot en las que se detecta un obstáculo.

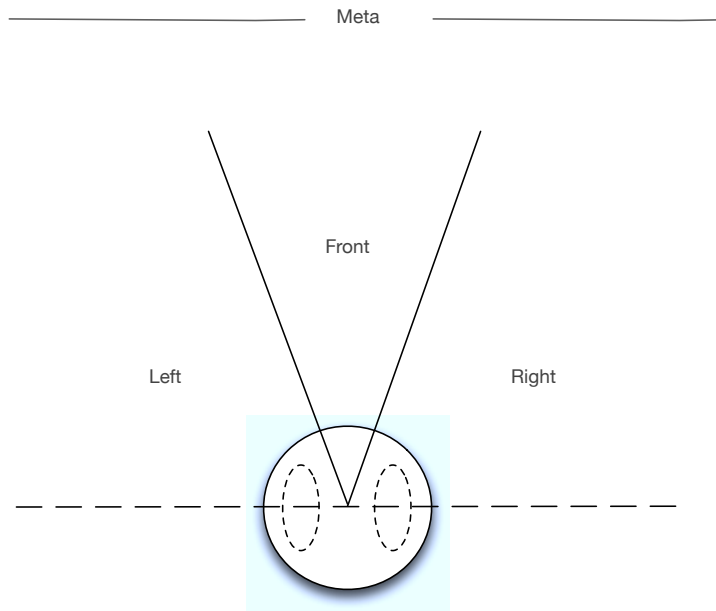
Si hay un obstáculo, la habilidad **CSkill\_SafeGoToPoint** bloquea la habilidad “Ir a un punto”. El robot evitará el obstáculo detectado activando la habilidad “Seguir el contorno”, una vez que lo haya superado bloqueará ésta habilidad y proseguirá con la ejecución de la habilidad “Ir a un punto”.

#### Casos posibles

A continuación mostramos algunos posibles casos que se le pueden presentar al robot durante la ejecución de esta habilidad y las acciones que se realizan en cada uno. La figura C.13, muestra las zonas donde es posible que el robot encuentre un obstáculo cercano y la posible posición de la meta (en cualquier lugar alrededor del robot).

La figura C.14, muestra los posibles casos donde la habilidad **CSkill\_SafeGoToPoint**, activará la habilidad “Seguir contorno” por la derecha del robot para rodear el lado izquierdo del obstáculo y así lograr evitarlo.

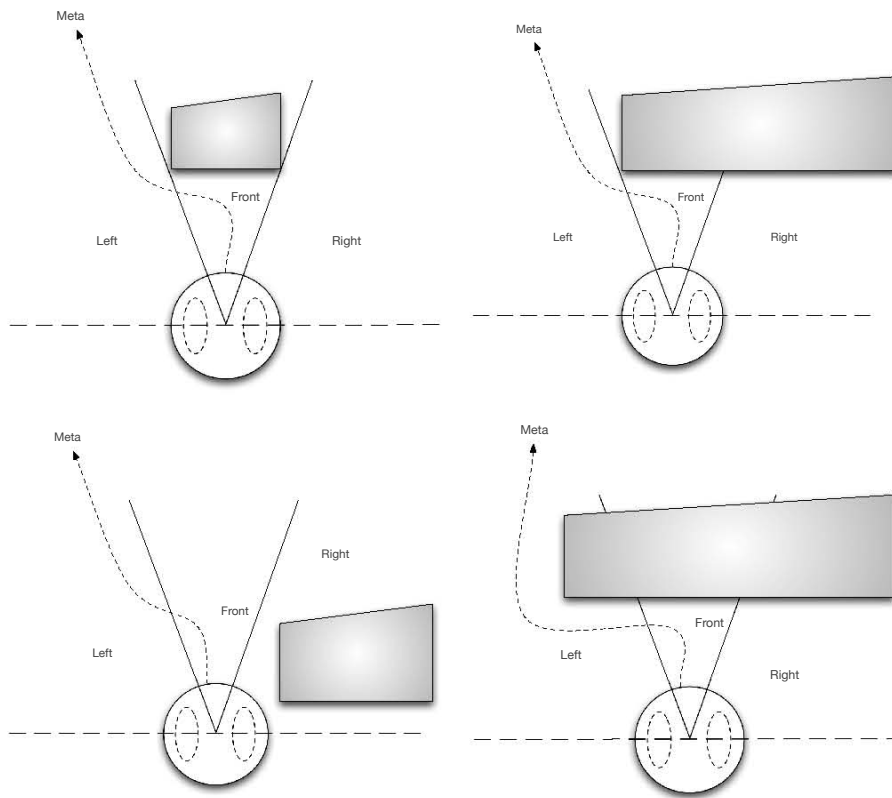




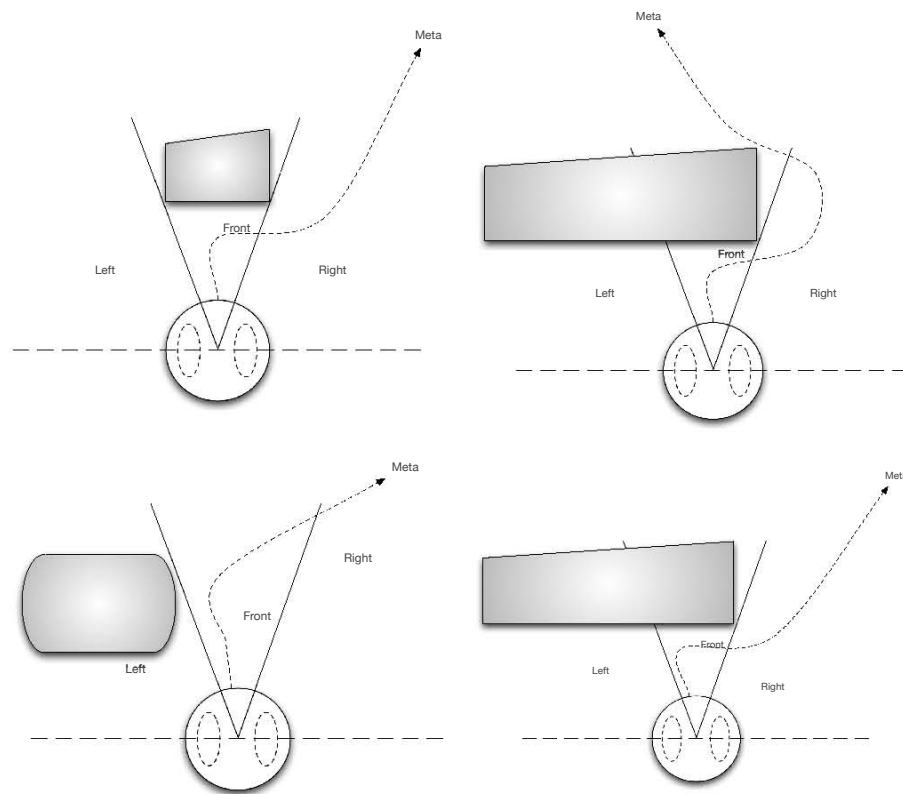
**Fig. C.13:** División de las zonas donde pueden ser detectados los obstáculos cercanos al robot.

La figura C.15, muestra los posibles casos donde la habilidad `CSkill_SafeGoToPoint`, activará la habilidad “Seguir contorno” por la izquierda del robot para rodear el lado izquierdo del obstáculo y así lograr evitarlo.

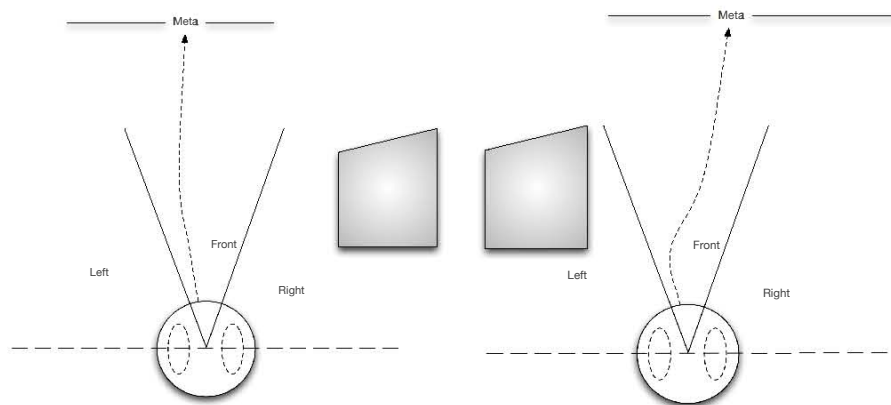
La figura C.16, muestra los posibles casos donde la habilidad `CSkill_SafeGoToPoint`, activará la habilidad “Ir a un punto”, ya que el robot tiene espacio libre para transitar. También se activará cuando no haya ningún obstáculo presente.



**Fig. C.14:** Casos en que se activará la habilidad “Seguir contorno” por la derecha del robot.



**Fig. C.15:** Casos en que se activará la habilidad “Seguir contorno” por la izquierda del robot.



**Fig. C.16:** Casos en que se activará la habilidad “Ir a un punto”.



# Bibliografía

- [Fol, 1979] (1979). *Architectural signing and graphics*. Whitney Library of Design.
- [Cos, 1987] (1987). *Señalética*. CEAC.
- [Avi, 2005] (2005). *Señalética. Conceptos y fundamentos. Una aplicación en bibliotecas*. Alfagrama Ediciones.
- [TN, 2006] (2006). Active tag rfid technovelgy. <http://www.technovelgy.com/>.
- [Sho, 2006] (2006). Rfid-driven shopping assistant robot soon to be tested in a mall. <http://ubiks.net/local/blog/jmt/archives3/004890.html>.
- [RFI, 2006] (2006). Tecnología rfid, introducción. RFID Magazine. <http://www.rfidmagazine.com>.
- [Onc, 2009] (2009). Accesibilidad y capacidades cognitivas: Movilidad en el entorno urbano vialidad, transporte y edificios públicos. Fundación ONCE y Technosite <http://accesibilidadcognitivaurbana.fundaciononce.es/>.
- [Allen, 1999] Allen, G. (1999). Cognitive abilities in the service of wayfinding: A functional approach. *The Professional Geographer*, 51(4):555–561.
- [Alonso-Martin and Salichs, 2011] Alonso-Martin, F. and Salichs, M. (2011). Integration of a voice recognition system in a social robot. *Cybernetics and Systems*, 42(4):215–245.
- [Barber, 2000] Barber, R. (2000). *Desarrollo de una arquitectura para robots móviles autónomos. Aplicación a un sistema de navegación topológica*. PhD thesis, Universidad Carlos III de Madrid.
- [Bekey, 2005] Bekey, G. A. (2005). *Autonomous Robots: From Biological Inspiration to Implementation and Control*. Intelligent Robots and autonomous agents. The MIT Press.
- [Berger, 2009] Berger, C. (2009). *Wayfinding: Designing and Implementing Graphic Navigational Systems*. A RotoVision book. RotoVision.
- [Bhatt and Glover, 2006] Bhatt, H. and Glover, B. (2006). *RFID Essentials*. O'Reilly.

- [Bhuptani and Moradpour, 2005] Bhuptani, M. and Moradpour, S. (2005). *RFID Field Guide. Deploying Radio Frequency Identification Systems*. Sun Microsystems and Prentice Hall.
- [Borenstein et al., 1994] Borenstein, J., Feng, L., and Everett, B. (1994). Where am i? sensors and methods for autonomous mobile robot localization. Technical report, The University of Michigan.
- [Caragliu et al., 2009] Caragliu, A., Del Bo, C., and Nijkamp, P. (2009). Smart cities in europe. Technical report.
- [Cheeseman et al., 1997] Cheeseman, P., Smith, R., and Self, M. (1997). A stochastic map for uncertain spatial relationships. In Press, M., editor, *4th International Symposium on Robotic Research*.
- [Conde, 2000] Conde, X. F. (2000). Introducción a la lingüística. *Ianua. Revista Philológica Románica*. Suplemento 01.
- [Cormen et al., 2001] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2001). *Introduction to Algorithms*. MIT Press and McGraw-Hill, 2 edition.
- [Corrales, 2008] Corrales, A. (2008). Sistema de identificación de objetos mediante rfid para un robot personal. Master's thesis, Universidad Carlos III de Madrid.
- [Corrales and Salichs, 2011] Corrales, A. and Salichs, M. (2011). Use of rfid technology on a mobile robot for topological navigation tasks. In *RFID-Technologies and Applications (RFID-TA), 2011 IEEE International Conference on*, pages 408–414.
- [Corrales and Salichs, 2009] Corrales, A. and Salichs, M. A. (2009). Integration of a rfid system in a social robot. In Kim, J.-H., Ge, S. S., Vadakkepat, P., Jesse, N., Al Manum, A., Puthusserypady K, S., Rückert, U., Sitte, J., Witkowski, U., Nakatsu, R., Braunl, T., Baltes, J., Anderson, J., Wong, C.-C., Verner, I., and Ahlgren, D., editors, *Progress in Robotics*, volume 44 of *Communications in Computer and Information Science*, pages 63–73. Springer Berlin Heidelberg.
- [Crocker and Overell, 2008] Crocker, D. and Overell, P. (2008). Official internet protocol standards. augmented bnf for syntax specifications: Abnf. <ftp://ftp.rfc-editor.org/in-notes/std/std68.txt>.
- [Dapena, 2002] Dapena, E. (2002). *Planificación de Caminos con incertidumbre para robots móviles*. PhD thesis, Universidad Carlos III de Madrid.
- [de Gorostiza, 2010] de Gorostiza, J. F. (2010). *Programación natural de un robot social mediante diálogos*. PhD thesis, Universidad Carlos III de Madrid.
- [Elmogy et al., 2008] Elmogy, M., Habel, C., and Zhang, J. (2008). Robot topological map generation from formal route instructions. In *The 6th International Cognitive Robotics Workshop at 18th European Conference on Artificial Intelligence (ECAI)*, pages 60–67, Patras, Greece. <http://ssrn.com/abstract=1835810> [doi:10.2139/ssrn.1835810](https://doi.org/10.2139/ssrn.1835810).

- [Elmogly et al., 2009] Elmogly, M., Habel, C., and Zhang, J. (2009). Spatial language for route-based humanoid robot navigation. *Cognitive Processing*, 10:208–211. 10.1007/s10339-009-0290-y.
- [Finkenzeller, 2004] Finkenzeller, K. (2004). *RFID Handbook*. WILEY, 2 edition.
- [Fox et al., 1999] Fox, D., Burgard, W., Dellaert, F., and Thrun, S. (1999). Monte carlo localization: Efficient position estimation for mobile robots. In *Proc. of the National Conference on Artificial Intelligence*.
- [Freundschuh, 1991] Freundschuh, S. M. (1991). *Spatial knowledge acquisition of urban environments from maps and navigation experience*. PhD thesis, State University of New York, Buffalo.
- [Gamma et al., 1994] Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional.
- [Gerkey et al., 2003] Gerkey, B., Vaughan, R. T., and Howard, A. (2003). The player/stage project: Tools for multi-robot and distributed sensor systems. In *11th International Conference on Advanced Robotics (ICAR 2003)*, pages 317–323, Coimbra, Portugal.
- [Golledge, 1999] Golledge, R. (1999). *Wayfinding Behavior. Cognitive mapping and other spatial processes*. The Johns Hopkins University Press.
- [Gorostiza and Salichs, 2011] Gorostiza, J. F. and Salichs, M. A. (2011). End-user programming of a social robot by dialog. *Robotics and Autonomous Systems*, 59(12):1102 – 1114.
- [Heuten et al., 2008] Heuten, W., Henze, N., Boll, S., and Pielot, M. (2008). Tactile wayfinder: a non-visual support system for wayfinding. In *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges, NordiCHI '08*, pages 172–181, New York, NY, USA. ACM.
- [Howard, 2006] Howard, A. (2006). Simple mapping utilities pmap. <http://www-robotics.usc.edu/~ahoward/pmap/index.html>. Robotics Research Lab Computer Science Department University of Southern California.
- [Huh et al., 2006] Huh, J., Lee, K., Chung, W. K., Jeong, W. S., and Kim, K. K. (2006). Mobile robot exploration in indoor environment using topological structure with invisible barcode. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 5265 –5272.
- [Huidobro, 2004] Huidobro, J. (2004). Rfid etiquetas inteligentes. *BIT Digital*, (146).
- [IEN, 2007] IEN (2007). *Accesibilidad universal : parte 1: criterios DALCO para facilitar la accesibilidad al entorno. Instituto Español de Normalización*. Norma española ; UNE 170001-1. AENOR.

- [IPv6style, 2006] IPv6style (2006). Ntt com and tmsuk conduct a field trial of robots that use ipv6 and rfid tags. <http://www.ipv6style.jp/en/special/20060213/index.shtml>.
- [Istomin and Dwyer, 2009] Istomin, K. V. and Dwyer, M. J. (2009). Finding the Way: A Critical Discussion of Anthropological Theories of Human Spatial Orientation with Reference to Reindeer Herders of Northeastern Europe and Western Siberia. *Current Anthropology*, 50(1):29–49.
- [Kim and Chong, 2007] Kim, M. and Chong, N. Y. (2007). Rfid-based mobile robot guidance to stationary target. *Mechatronics Science Direct*, 17(1):217–229.
- [Kulyukin et al., 2005] Kulyukin, V., Gharpure, C., and Nicholson, J. (2005). Robo-cart: toward robot-assisted navigation of grocery stores by the visually impaired. In *Intelligent Robots and Systems, 2005. (IROS 2005)*, pages 2845 – 2850.
- [Kulyukin et al., 2006] Kulyukin, V., Gharpure, C., Nicholson, J., and Osborne, G. (2006). Robot-assisted wayfinding for the visually impaired in structured indoor environments. *Auton. Robots*, 21(1):29–41.
- [Kulyukin et al., 2008] Kulyukin, V., Kutiyanawala, A., LoPresti, E., Matthews, J., and Simpson, R. (2008). iwalker: Toward a rollator-mounted wayfinding system for the elderly. In *RFID, 2008 IEEE International Conference on*, pages 303 – 311.
- [Lee and Song, 2007] Lee, S. and Song, J.-B. (2007). Use of coded infrared light as artificial landmarks for mobile robot localization. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1731 – 1736.
- [Liu et al., 2009] Liu, A. L., Hile, H., Borriello, G., Brown, P. A., Harniss, M., Kautz, H., and Johnson, K. (2009). Customizing directions in an automated wayfinding system for individuals with cognitive impairment. In *Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility*, Assets '09, pages 27–34, New York, NY, USA. ACM.
- [Lynch, 1960] Lynch, K. (1960). *The Image of the City*. MIT Press, Cambridge MA.
- [Mata et al., 2001] Mata, M., Armingol, J., de la Escalera, A., and Salichs, M. (2001). A visual landmark recognition system for topological navigation of mobile robots. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 2, pages 1124 – 1129 vol.2.
- [Montemello, 2005] Montemello, D. (2005). *The Cambridge Handbook of Visuospatial Thinking*, chapter Navigation. Cambridge University Press.
- [Moon et al., 1999] Moon, I., Miura, J., and Shirai, Y. (1999). Online selection of stable visual landmarks under uncertainty. In *Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on*, volume 2, pages 781 – 786 vol.2.



- [Muhlhausen, 2000] Muhlhausen, J. (2000). Wayfinding is not signage. Signs of the Times <http://www.signweb.com/index.php/channel/6/id/1433/>.
- [Murphy, 2000] Murphy, R. R. (2000). *Introduction to AI Robotics*. Intelligent Robots and autonomous agents. The MIT Press.
- [Rivas, 2010] Rivas, R. (2010). *Diseño Software de una Arquitectura de Control de Robots Autónomos Inteligentes. Aplicación a un Robot Social*. PhD thesis, Universidad Carlos III de Madrid, Leganés España.
- [Rivas et al., 2007] Rivas, R., Corrales, A., Barber, R., and M.A.Salichs (2007). Robot skill abstraction for ad architecture. In *6th IFAC Symposium on Intelligent Autonomous Vehicles IAV*, Toulouse Francia. IFAC.
- [Salichs et al., 2009] Salichs, J., Castro-González, A., and Salichs, M. A. (2009). Infrared remote control with a social robot. In Kim, J.-H., Ge, S. S., Vadakkepat, P., Jesse, N., Al Manum, A., Puthusserypady K, S., Rückert, U., Sitte, J., Witkowski, U., Nakatsu, R., Braunl, T., Baltes, J., Anderson, J., Wong, C.-C., Verner, I., and Ahlgren, D., editors, *Progress in Robotics*, volume 44 of *Communications in Computer and Information Science*, pages 86–95. Springer Berlin Heidelberg.
- [Salichs, 2001] Salichs, M. (2001). Navigation of mobile robots: Learning from human beings. IFAC Workshop on Mobile Robot Technology.
- [Salichs and Moreno, 2000] Salichs, M. A. and Moreno, L. (2000). Navigation of mobile robots: Open questions. *Robotics*, 18:227–234.
- [Salichs et al., 2006] Salichs, M. A., R.Barber, A.M.Khamis, M.Malfaz, J.F.Gorostiza, R.Pacheco, R.Rivas, A.Corrales, and E.Delgado (2006). Maggie: A robotic platform for human-robot social interaction. In IEEE, editor, *IEEE International Conference on Robotics, Automation and Mechatronics*, Bangkok. Thailand.
- [Sandip, 2006] Sandip, L. (2006). *RFID sourcebook*. IBM Press.
- [Shepard, 2005] Shepard, S. (2005). *RFID Radio Frequency Identification*. McGraw-Hill.
- [Siegwart and Nourbakhsh, 2004] Siegwart, R. and Nourbakhsh, I. (2004). *Introduction to Autonomous Mobile Robots*. MIT Press, bradford book edition.
- [Sorrrows and Hirtle, 1999] Sorrows, M. E. and Hirtle, S. C. (1999). The nature of landmarks for real and electronic spaces. In *Proceedings of the International Conference on Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science*, COSIT '99, pages 37–50, London, UK. Springer-Verlag.

- [Tenbrink et al., 2010] Tenbrink, T., Ross, R. J., Thomas, K. E., Dethlefs, N., and Andonova, E. (2010). Route instructions in map-based human-human and human-computer dialogue: A comparative analysis. *J. Vis. Lang. Comput.*, 21:292–309.
- [Thrun et al., 2005] Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics*. The MIT Press.
- [Tolman, 1948] Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review*, 55:189–208.
- [Tuikka et al., 2011] Tuikka, T., Siira, E., and Saukko, M. (2011). City service discovery and access with near field communication. In *Information Science and Service Science (NISS), 2011 5th International Conference on New Trends in*, volume 1, pages 116 –121.
- [W3C, 2004] W3C (2004). Speech recognition grammar specification. <http://www.w3.org/TR/speech-grammar/>.
- [Wang and Ishimatsu, 2005] Wang, H. and Ishimatsu, T. (2005). Vision-based navigation for an electric wheelchair using ceiling light landmark. *Journal of Intelligent and Robotic Systems*, 41:283–314. 10.1007/s10846-005-9902-7.
- [Weiss et al., 2010] Weiss, A., Igelsbock, J., Tscheligi, M., Bauer, A., Kuhnlenz, K., Wollherr, D., and Buss, M. (2010). Robots asking for directions; the willingness of passers-by to support robots. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, pages 23 –30.
- [Wen et al., 2009] Wen, F., Yuan, K., Zou, W., Chai, X., and Zheng, R. (2009). Visual navigation of an indoor mobile robot based on a novel artificial landmark system. In *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, pages 3775 –3780.
- [Wiener et al., 2009] Wiener, J. M., Buchner, S. J., and Holscher, C. (2009). Taxonomy of human wayfinding tasks: A knowledge-based approach. *Spatial Cognition and Computation: An Interdisciplinary Journal*, 9(2):152–165.
- [Willis and Helal, 2005] Willis, S. and Helal, S. (2005). Rfid information grid for blind navigation and wayfinding. In *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, pages 34 – 37.
- [Xia et al., 2008] Xia, J. C., Arrowsmith, C., Jackson, M., and Cartwright, W. (2008). The wayfinding process relationships between decision-making and landmark utility. *Tourism Management*, 29(3):445 – 457.
- [Yang et al., 2011] Yang, R., Park, S., Mishra, S. R., Hong, Z., Newsom, C., Joo, H., Hofer, E., and Newman, M. W. (2011). Supporting spatial awareness and independent wayfinding for pedestrians with visual impairments. In *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility, ASSETS '11*, pages 27–34, New York, NY, USA. ACM.

- [Yébenes et al., 2011] Yébenes, M., Corrales, A., and M.A.Salichs (2011). Ampliación de los sistemas rfid en el robot maggie. Technical report, Universidad Carlos III de Madrid. Departamento de Ingeniería de Sistemas y Automática. Proyecto Fin de Carrera. Titulación: Ingeniería de Telecomunicación.